



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KS 141501

OPTIMASI ALOKASI PRODUK PADA RAK DISPLAI MINIMARKET BENTAR MENGGUNAKAN ALGORITMA GENETIKA YANG DIMODIFIKASI

LISTYANI NIRMALASARI
NRP 5211 100 072

Dosen Pembimbing
Wiwik Anggraeni, S.Si., M.Kom.

JURUSAN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2015



FINAL PROJECT - KS 141501

***OPTIMIZATION PRODUCT ALLOCATION ON BENTAR
MINIMARKET'S DISPLAY SHELF USING MODIFIED
GENETIC ALGORITHM***

LISTYANI NIRMALASARI
NRP 5211 100 072

Supervisor
Wiwik Anggraeni, S.Si., M.Kom.

DEPARTMENT OF INFORMATION SYSTEM
Faculty of Information Technology
Institute of Technology Sepuluh Nopember
Surabaya 2015

LEMBAR PENGESAHAN

OPTIMASI ALOKASI PRODUK PADA RAK DISPLAI MINIMARKET BENTAR MENGGUNAKAN ALGORITMA GENETIKA YANG DIMODIFIKASI

TUGAS AKHIR

Disusun untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Jurusan Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

LISTYANI NIRMALASARI

5211 100 072

Surabaya, Juli 2015

**KETUA
JURUSAN SISTEM INFORMASI**

Dr. Eng. Febrilyan Samopa, S.Kom, M.Kom

NIP 19730219 199802 1 001

JURUSAN
SISTEM INFORMASI

LEMBAR PERSETUJUAN

OPTIMASI ALOKASI PRODUK PADA RAK DISPLAI MINIMARKET BENTAR MENGGUNAKAN ALGORITMA GENETIKA YANG DIMODIFIKASI

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
Pada
Jurusan Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

LISTYANI NIRMALASARI

5211 100 072

Disetujui Tim Penguji : Tanggal Ujian : 22 Juni 2015
Periode Wisuda : September 2015

Wiwik Anggraeni, S.Si., M.Kom.

(Pembimbing I)

Bekti Cahyo Hidayanto, S.Si., M.Kom.

(Penguji 1)

Renny Pradina, S.T., M.T.

(Penguji 2)

OPTIMASI ALOKASI PRODUK PADA RAK DISPLAI MINIMARKET BENTAR MENGGUNAKAN ALGORITMA GENETIKA YANG DIMODIFIKASI

Nama Mahasiswa : Listyani Nirmalasari
NRP : 5211 100 072
Jurusan : Sistem Informasi FTIf-ITS
Dosen Pembimbing : Wiwik Anggraeni, S.Si., M.Kom.

ABSTRAK

Bisnis ritel di Indonesia saat ini sedang berkembang pesat. Salah satu faktor kesuksesan bisnis ritel didukung oleh alokasi produk yang efisien. Alokasi produk pada ruang rak display merupakan satu aspek penting dalam bisnis retail. Susunan produk pada rak memiliki dampak pada penjualan di beberapa produk retail. Keterbatasan ruang rak membuat peritel harus teliti dalam mengalokasikan setiap produk dan dimana meletakkannya di rak sehingga mampu memberikan keuntungan yang maksimal.

Pada optimasi alokasi produk pada ruang rak display minimarket kali ini akan digunakan algoritma genetika yang dimodifikasi dengan pemrograman integer untuk mencari solusi dari semua kemungkinan yang ada. Metode ini dapat digunakan untuk menyelesaikan permasalahan optimasi alokasi produk. Pemrograman integer pada metode ini berfungsi untuk mendapatkan nilai bulat dari hasil algoritma genetika yang bernilai riil.

Hasil dari penelitian ini adalah konfigurasi jumlah alokasi produk kategori susu bubuk pada rak display minimarket yang dibagi dalam lima layer. Jumlah alokasi produk tersebut digunakan untuk membuat peta konfigurasi produk di rak display. Peta konfigurasi produk ini dapat membantu peritel dalam menentukan jumlah produk yang harus didisplai pada rak sehingga alokasi produk yang didisplai dapat optimal dan total keuntungan yang diperoleh pada setiap layer yaitu Rp

30.015,00 untuk layer 1; Rp 46.514,00 untuk layer 2; Rp 50.473,75 untuk layer 3; Rp 44.465,00 untuk layer 4; dan Rp 54.910,00 untuk layer 5.

Kata kunci : *Alokasi produk, Algoritma Genetika, Optimasi, Ritel, Rak display*

OPTIMIZATION PRODUCT ALLOCATION ON BENTAR MINIMARKET'S DISPLAY SHELF USING MODIFIED GENETIC ALGORITHM

Name : Listyani Nirmalasari
Student's Reg. Number : 5211 100 072
Department : Information Systems FTIf - ITS
Supervisors : Wiwik Anggraeni, S.Si., M.Kom.

ABSTRACT

Retail business is growing rapidly in Indonesia. One of the success factors of retail businesses are supported by the efficiency of allocation products. Allocation on shelf display is an important aspect of the retail business. The composition of the product on the shelf have an impact on retail sales in some products. Shelf space imitations make retailers should be careful in allocating in each product and where to put them on the shelf thus it can provide maximum benefit.

This shelf-space allocation optimization will use a modified genetic algorithm with Integer Programming to look for a solution of all the possibilities that exist. This method can be used to solve optimization of product allocation problems. Integer Programming is used to get the integer value from genetic algorithm's value.

The result of this study is the conguration of children milk powder product's allocation on the minimarket shelf-diplay which is divided into five layers. Total amount of allocation products are suded to make the children milk powder product's placement map. The placement map can help retailer in determining the amount of product that should displayed so that it can optimized and the total profit that earned from each layers are Rp 30.015,00 for first layer; Rp 46.514,00 for second layer; Rp 50.473,75 for third layer; Rp 44.465,00 for fourth layer; and Rp 54.910,00 for fifth layer.

Keywords: *Allocation product, Genetic Algorithm, Optimization, Retail, Shelf-display*

KATA PENGANTAR

Puji syukur senantiasa dihaturkan kepada Yang Maha Memberi Hidup Allah SWT. yang telah senantiasa memberikan rahmat dan hidayah kepada kita semua baik jasmani maupun rohani. Allah SWT. yang selalu memberikan kekuatan luar biasa sehingga tugas akhir dengan judul **“Optimasi Alokasi Produk Pada Rak Displai Minimarket Bentar Menggunakan Algoritma Genetika Yang Dimodifikasi”** dapat terselesaikan dengan baik. Sebagai salah satu syarat kelulusan meraih gelar Sarjana di Jurusan Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya. Dengan terselesaikannya tugas akhir ini, maka berakhir pula masa studi penulis di kampus tercinta ini.

Tugas Akhir ini dilaksanakan pada bulan Oktober 2014 hingga Juni 2015 bertempat di Minimarket Bentar cabang Delta Sari, Sidoarjo.

Penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada pihak-pihak yang telah membantu dalam proses pengerjaan tugas akhir ini, diantaranya :

- Bapak Kamid sebagai pemilik dari Minimarket Bentar. Terima kasih telah mengizinkan penulis untuk melakukan penelitian di Minimarket Bentar cabang Sidoarjo.
- Mas Reza dan Ibu Dewi selaku supervisor Minimarket Bentar. Terima kasih telah mengizinkan penulis untuk mengambil data penelitian di Minimarket Bentar cabang Delta Sari Sidoarjo.
- Bapak Dr.Eng. Febriliyan Samopa, S.Kom., M.Kom., selaku Kepala Jurusan Sistem Informasi, Institut Teknologi Sepuluh Nopember Surabaya.
- Bapak Tony Dwi Susanto, S.T., M.T., Ph.D selaku Ketua Program Studi S1 Jurusan Sistem Informasi, Institut Teknologi Sepuluh Nopember Surabaya.
- Ibu Wiwik Anggraeni, S.Si, M.Kom., selaku Dosen Pembimbing tugas akhir penulis. Terima kasih telah

membantu dan memotivasi dalam menyelesaikan tugas akhir dan mencapai kelulusan.

Penulis sadar bahwa masih banyak kekurangan yang perlu diperbaiki dari tugas akhir ini. Oleh karena itu, penulis berharap agar tugas akhir ini dapat dikembangkan di waktu yang akan datang. Akhir kata, semoga tugas akhir ini memberikan manfaat kepada para pembaca.

Surabaya, Juni 2015

Penulis

DAFTAR ISI

ABSTRAK.....	V
ABSTRACT.....	VII
KATA PENGANTAR	IX
DAFTAR ISI.....	XI
DAFTAR GAMBAR	XV
DAFTAR TABEL.....	XVII
DAFTAR FORMULA	XXI
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah.....	3
1.3 Batasan Tugas Akhir	3
1.4 Tujuan Tugas Akhir	4
1.5 Manfaat Tugas Akhir	4
1.6 Sistematika Penulisan.....	4
BAB II TINJAUAN PUSTAKA.....	7
2.1 Studi Sebelumnya.....	7
2.2 Dasar Teori.....	10
2.2.1 Minimarket Bentar	10
2.2.2 <i>Retailing</i>	12
2.2.3 Manajemen Ruang <i>Display</i>	15
2.2.4 Algoritma Genetika.....	16
2.2.5 <i>Integer Programming</i>	19
BAB III METODOLOGI PENELITIAN	23
3.1 Studi literatur.....	24
3.2 Pengumpulan data	25
3.3 Pengolahan Data dan Analisis.....	25
3.3.1 Pendefinisian Model Optimasi	25

3.3.2	Pengalokasian Produk pada Rak	27
3.3.3	Analisis Hasil Pengolahan Data	27
3.4	Pembuatan Kesimpulan dan Saran	27
3.5	Penyusunan Laporan Tugas Akhir.....	28
BAB IV PERANCANGAN		29
4.1	Pengolahan Data	29
4.1.1	Data Masukan.....	29
4.1.2	Data Proses	29
4.1.3	Data Keluaran.....	29
4.2	Pembuatan Model Optimasi	30
4.2.1	Pendefinisian Variabel Keputusan	30
4.2.2	Formulasi Fungsi Tujuan.....	33
4.2.3	Formulasi Fungsi Batasan	35
4.3	Rancangan Algoritma Genetika.....	38
BAB V IMPLEMENTASI		43
5.1	Implementasi Algoritma Genetika.....	43
5.2	Implementasi Pemrograman Integer.....	52
BAB VI HASIL DAN PEMBAHASAN		55
6.1	Lingkungan Uji Coba	55
6.2	Parameter Uji Coba	56
6.3	Skenario Uji Coba	56
6.4	Hasil Uji Coba	57
6.4.1	Perbandingan Total Keuntungan Berdasarkan Parameter Nilai N_{pop} 30, 50, Dan 100 Pada Layer 1	57
6.4.2	Perbandingan Total Keuntungan Berdasarkan Parameter P_c Antara 0.1 – 0.9 Pada Layer 1 ...	59
6.4.3	Perbandingan Total Keuntungan Berdasarkan Fungsi Mutasi dan Probabilitas Mutasi 0,1 dan 0,05 Pada Layer 1	59
6.4.4	Perbandingan Total Keuntungan Berdasarkan Penggunaan Fungsi <i>Hybrid Fmincon</i> Pada Layer 1	61

6.4.5	Perbandingan Total Keuntungan Berdasarkan Parameter Nilai N_{pop} 30, 50, Dan 100 Pada Layer 2	63
6.4.6	Perbandingan Total Keuntungan Berdasarkan Parameter P_c Antara 0.1 – 0.9 Pada Layer 2 ...	64
6.4.7	Perbandingan Total Keuntungan Berdasarkan Fungsi Mutasi dan Probabilitas Mutasi 0,1 dan 0,05 Pada Layer 2.....	65
6.4.8	Perbandingan Total Keuntungan Berdasarkan Penggunaan Fungsi <i>Hybrid Fmincon</i> Pada Layer 2	66
6.4.9	Perbandingan Total Keuntungan Berdasarkan Parameter Nilai N_{pop} 30, 50, Dan 100 Pada Layer 3	68
6.4.10	Perbandingan Total Keuntungan Berdasarkan Parameter P_c Antara 0.1 – 0.9 Pada Layer 3 ...	69
6.4.11	Perbandingan Total Keuntungan Berdasarkan Fungsi Mutasi dan Probabilitas Mutasi 0,1 dan 0,05 Pada Layer 3.....	70
6.4.12	Perbandingan Total Keuntungan Berdasarkan Penggunaan Fungsi <i>Hybrid Fmincon</i> Pada Layer 3	72
6.4.13	Perbandingan Total Keuntungan Berdasarkan Parameter Nilai N_{pop} 30, 50, Dan 100 Pada Layer 4	73
6.4.14	Perbandingan Total Keuntungan Berdasarkan Parameter P_c Antara 0.1 – 0.9 Pada Layer 4 ...	75
6.4.15	Perbandingan Total Keuntungan Berdasarkan Fungsi Mutasi dan Probabilitas Mutasi 0,1 dan 0,05 Pada Layer 4.....	75
6.4.16	Perbandingan Total Keuntungan Berdasarkan Penggunaan Fungsi <i>Hybrid Fmincon</i> Pada Layer 4	77
6.4.17	Perbandingan total keuntungan berdasarkan parameter nilai N_{pop} 30, 50, dan 100 pada Layer 5.....	78

6.4.18	Perbandingan Total Keuntungan Berdasarkan Parameter P_c Antara 0.1 – 0.9 Pada Layer 5 ...	80
6.4.19	Perbandingan Total Keuntungan Berdasarkan Fungsi Mutasi dan Probabilitas Mutasi 0,1 dan 0,05 Pada Layer 5	80
6.4.20	Perbandingan Total Keuntungan Berdasarkan Penggunaan Fungsi <i>Hybrid Fmincon</i> Pada Layer 5	82
6.5	Penerapan Hasil Optimal Algoritma Genetika pada Pemrograman Integer	84
6.6	Validasi	91
6.7	Analisis Hasil	94
6.7.1	Analisis Hasil Uji Coba Parameter	94
6.7.2	Analisis Perbandingan Hasil Algoritma Genetika dan Algoritma Genetika yang Dimodifikasi	96
6.7.3	Analisis Hasil Keseluruhan dan Konfigurasi Produk pada Rak <i>Display</i>	99
BAB VII	KESIMPULAN DAN SARAN	103
7.1	Kesimpulan	103
7.2	Saran	104
DAFTAR PUSTAKA	105
LAMPIRAN A	A-1
LAMPIRAN B	B-1
LAMPIRAN C	C-1

DAFTAR TABEL

Tabel 2. 1 Studi Sebelumnya.....	7
Tabel 6. 1 Lingkungan uji coba perangkat keras	55
Tabel 6. 2 Lingkungan uji coba perangkat lunak	55
Tabel 6. 3 Uji coba $N_{pop} = 30$ pada Layer 1	57
Tabel 6. 4 Uji coba $N_{pop} = 30$ pada Layer 1	58
Tabel 6. 5 Uji coba $N_{pop} = 30$ pada Layer 1	58
Tabel 6. 6 Uji coba parameter P_m dengan $N_{pop} = 30$ pada Layer 1	60
Tabel 6. 7 Uji coba parameter P_m dengan $N_{pop} = 50$ pada Layer 1	60
Tabel 6. 8 Uji coba parameter P_m dengan $N_{pop}=100$ pada Layer 1	60
Tabel 6. 9 Uji coba penggunaan f_{mincon} dengan $N_{pop}=30$ pada Layer 1	61
Tabel 6. 10 Uji coba penggunaan f_{mincon} dengan $N_{pop}=50$ pada Layer 1	62
Tabel 6. 11 Uji coba penggunaan f_{mincon} dengan $N_{pop}=100$ pada Layer 1	62
Tabel 6. 12 Uji coba $N_{pop} = 30$ pada Layer 2	63
Tabel 6. 13 Uji coba $N_{pop} = 50$ pada Layer 2	63
Tabel 6. 14 Uji coba $N_{pop} = 100$ pada Layer 2	63
Tabel 6. 15 Uji coba parameter P_m dengan $N_{pop} = 30$ pada Layer 2	65
Tabel 6. 16 Uji coba parameter P_m dengan $N_{pop} = 50$ pada Layer 2	65
Tabel 6. 17 Uji coba parameter P_m dengan $N_{pop} = 100$ pada Layer 2	66
Tabel 6. 18 Uji coba penggunaan f_{mincon} dengan $N_{pop}=30$ pada Layer 2	67
Tabel 6. 19 Uji coba penggunaan f_{mincon} dengan $N_{pop}=50$ pada Layer 2	67
Tabel 6. 20 Uji coba penggunaan f_{mincon} dengan $N_{pop}=100$ pada Layer 2	67

Tabel 6. 21 Uji coba $N_{pop} = 30$ pada Layer 3	68
Tabel 6. 22 Uji coba $N_{pop} = 50$ pada Layer 3	69
Tabel 6. 23 Uji coba $N_{pop} = 100$ pada Layer 3	69
Tabel 6. 24 Uji coba parameter P_m dengan $N_{pop}= 30$ pada Layer 3	70
Tabel 6. 25 Uji coba parameter P_m dengan $N_{pop} = 50$ pada Layer 3	71
Tabel 6. 26 Uji coba parameter P_m dengan $N_{pop} = 100$ pada Layer 3	71
Tabel 6. 27 Uji coba penggunaan $fmincon$ dengan $N_{pop}=30$ pada Layer 3.....	72
Tabel 6. 28 Uji coba penggunaan $fmincon$ dengan $N_{pop}=50$ pada Layer 3.....	72
Tabel 6. 29 Uji coba penggunaan $fmincon$ dengan $N_{pop}=100$ pada Layer 3.....	73
Tabel 6. 30 Uji coba $N_{pop} = 30$ pada Layer 4	73
Tabel 6. 31 Uji coba $N_{pop} = 50$ pada Layer 4	74
Tabel 6. 32 Uji coba $N_{pop} = 100$ pada Layer 4	74
Tabel 6. 33 Uji coba parameter P_m dengan $N_{pop} = 30$ pada Layer 4	76
Tabel 6. 34 Uji coba parameter P_m dengan $N_{pop} = 50$ pada Layer 4	76
Tabel 6. 35 Uji coba parameter P_m dengan $N_{pop} = 100$ pada Layer 4	76
Tabel 6. 36 Uji coba penggunaan $fmincon$ dengan $N_{pop}=30$ pada Layer 4.....	77
Tabel 6. 37 Uji coba penggunaan $fmincon$ dengan $N_{pop}=50$ pada Layer 4.....	77
Tabel 6. 38 Uji coba penggunaan $fmincon$ dengan $N_{pop} = 100$ pada Layer 4.....	78
Tabel 6. 39 Uji coba $N_{pop} = 30$ pada Layer 5	79
Tabel 6. 40 Uji coba $N_{pop} = 50$ pada Layer 5	79
Tabel 6. 41 Uji coba $N_{pop} = 100$ pada Layer 5	79
Tabel 6. 42 Uji coba parameter P_m dengan $N_{pop}=30$ pada Layer 5	81
Tabel 6. 43 Uji coba parameter P_m dengan $N_{pop}=50$ pada Layer 5	81

Tabel 6. 44 Uji coba parameter P_m dengan $N_{pop}=100$ pada Layer 5	81
Tabel 6. 45 Uji coba penggunaan <i>fmincon</i> dengan $N_{pop}=30$ pada Layer 5	82
Tabel 6. 46 Uji coba penggunaan <i>fmincon</i> dengan $N_{pop}=50$ pada Layer 5	83
Tabel 6. 47 Uji coba penggunaan <i>fmincon</i> dengan $N_{pop} = 100$ pada Layer 5	83
Tabel 6. 48 Hasil Optimal Algoritma Genetika pada Layer 1	84
Tabel 6. 49 Hasil Optimal Algoritma Genetika pada Layer 2	85
Tabel 6. 50 Hasil Optimal Algoritma Genetika pada Layer 3	87
Tabel 6. 51 Hasil Optimal Algoritma Genetika pada Layer 4	88
Tabel 6. 52 Hasil Optimal Algoritma Genetika pada Layer 5	89
Tabel 6. 53 Perbandingan Hasil Algoritma Genetika dan Algoritma Genetika-Pemrograman Integer pada Layer 1	96
Tabel 6. 54 Perbandingan Hasil Algoritma Genetika dan Algoritma Genetika-Pemrograman Integer pada Layer 2	96
Tabel 6. 55 Perbandingan Hasil Algoritma Genetika dan Algoritma Genetika-Pemrograman Integer pada Layer 3	97
Tabel 6. 56 Perbandingan Hasil Algoritma Genetika dan Algoritma Genetika-Pemrograman Integer pada Layer 4	98
Tabel 6. 57 Perbandingan Hasil Algoritma Genetika dan Algoritma Genetika-Pemrograman Integer pada Layer 5	98
Tabel 6. 58 Hasil Optimal	99

Halaman ini sengaja dikosongkan

DAFTAR GAMBAR

Gambar 2. 1 Rak Berbentuk <i>Island Gondola</i>	11
Gambar 2. 2 Proses <i>Supply Chain</i>	12
Gambar 2. 3 <i>Flow Diagram</i> Algoritma Genetika (Sumber: Turban, 2010)	18
Gambar 2. 4 <i>Flowchart</i> Algoritma Genetika Kontinu (Sumber: Haupt, 2004)	19
Gambar 2. 5 Pohon Enumerasi pada Branch and Bound	22
Gambar 3. 1 Metodologi Pengerjaan Tugas Akhir.....	23
Gambar 3. 2 Ilustrasi Rak Displai	26
Gambar 4. 1 Proses Algoritma Genetika pada <i>toolbox ga</i>	38
Gambar 5. 1 Memanggil <i>toolbox optimasi ga</i>	43
Gambar 5. 2 Tampilan <i>toolbox optimasi ga</i>	43
Gambar 5. 3 <i>Problem</i>	44
Gambar 5. 4 <i>Constraints</i>	46
Gambar 5. 5 <i>Run Solver and view results</i>	47
Gambar 5. 6 <i>Population</i>	48
Gambar 5. 7 <i>Fitness scalling</i>	48
Gambar 5. 8 <i>Selection</i>	49
Gambar 5. 9 <i>Reproduction</i>	49
Gambar 5. 10 <i>Mutation</i>	50
Gambar 5. 11 <i>Crossover</i>	50
Gambar 5. 12 <i>Hybrid function</i>	51
Gambar 5. 13 <i>Plot functions</i>	51
Gambar 5. 14 <i>Display to command window</i>	52
Gambar 5. 15 Inisiliasasi POM-QM	52
Gambar 5. 16 Membuat <i>data set</i> pada modul <i>integer & mixed</i> <i>integer programming</i>	53
Gambar 5. 17 <i>Data set</i> permasalahan pada rak - layer 1	54
Gambar 5. 18 <i>Output</i> dari <i>integer programming</i>	54
Gambar 6. 1 Hasil algoritma genetika dan pemrograman integer pada Layer 1	85
Gambar 6. 2 Hasil algoritma genetika dan pemrograman integer pada Layer 2	86

Gambar 6. 3 Hasil algoritma genetika dan pemrograman integer pada Layer 3.....	88
Gambar 6. 4 Hasil algoritma genetika dan pemrograman integer pada Layer 4.....	89
Gambar 6. 5 Hasil algoritma genetika dan pemrograman integer pada Layer 5.....	90
Gambar 6. 6 Hasil keluaran QM pada Layer 1	91
Gambar 6. 7 Hasil keluaran QM pada Layer 2	92
Gambar 6. 8 Hasil keluaran QM pada Layer 3	92
Gambar 6. 9 Hasil keluaran QM pada Layer 4	93
Gambar 6. 10 Hasil keluaran QM pada Layer 5	93
Gambar 6. 11 Perbandingan Total Keuntungan Saat Ini dan Hasil Optimasi	100
Gambar 6. 12 Konfigurasi Rak <i>Display</i>	101

DAFTAR FORMULA

Formula 5. 1 <i>Fitness function</i> rak - Layer 1	44
Formula 5. 2 <i>Fitness function</i> rak - Layer 2	44
Formula 5. 3 <i>Fitness function</i> rak - Layer 3	45
Formula 5. 4 <i>Fitness function</i> rak - Layer 4	45
Formula 5. 5 <i>Fitness function</i> rak - Layer 5	45
Formula 5. 6 <i>Nonlinear constraint function</i> pada Layer 1	46
Formula 5. 7 <i>Nonlinear constraint function</i> pada Layer 2	46
Formula 5. 8 <i>Nonlinear constraint function</i> pada Layer 3	46
Formula 5. 9 <i>Nonlinear constraint function</i> pada Layer 4	46
Formula 5. 10 <i>Nonlinear constraint function</i> pada Layer 5	47

Halaman ini sengaja dikosongkan

BAB I

PENDAHULUAN

Bab ini membahas tentang latar belakang, perumusan masalah, batasan, tujuan, manfaat dan sistematika penulisan tugas akhir.

1.1 Latar Belakang

Bisnis ritel di Indonesia saat ini berkembang pesat diikuti kemajuan perekonomian Indonesia. Menurut Pudjianto, Ketua Umum Asosiasi Pengusaha Ritel Indonesia (Aprindo), industri ritel di Indonesia terus mengalami pertumbuhan minimal sekitar 10 persen/tahun [1]. Peningkatan ini ditunjukkan dengan angka pertumbuhan ritel tradisional mencapai 42% dan ritel modern sebanyak 400% [2]. Salah satu penyebab bertumbuhnya bisnis ritel ini adalah meningkatnya daya beli masyarakat dari tahun ke tahun. Peningkatan daya beli ini menyebabkan berubahnya perilaku konsumen ke arah produk-produk mewah yang berdampak positif bagi bisnis ritel, terutama ritel modern.

Ritel modern memberikan alternatif belanja menarik yang menawarkan pelayanan dan kualitas produk dengan harga yang bersaing dengan pasar tradisional. Terdapat tiga jenis ritel modern di Indonesia yaitu minimarket, supermarket, dan hypermarket. Minimarket menjual barang-barang kebutuhan sehari-hari secara eceran dan langsung kepada konsumen akhir. Pada tahun 2004 – 2008, omset Minimarket meningkat dengan rata-rata 38,1% per tahun [3]. Data Lembaga riset Nielsen menunjukkan pertumbuhan minimarket sepanjang tahun 2010 mencapai 16.922 gerai dibanding tahun 2009 yang mencapai 11.927 gerai. Angka ini akan terus bertambah disebabkan oleh beberapa faktor. Faktor pertambahan jumlah penduduk di Indonesia dapat meningkatkan jumlah pelanggan yang potensial bagi minimarket. Faktor lain yang mempengaruhi

adalah adanya peningkatan pendapat per kapita serta perilaku masyarakat dalam berbelanja, khususnya masyarakat perkotaan.

Pesatnya pertumbuhan gerai di Indonesia, menimbulkan persaingan yang semakin ketat antar minimarket. Hal ini yang menuntut perusahaan ritel untuk menerapkan suatu manajemen ritel. Manajemen ritel diterapkan untuk meningkatkan penjualan dengan mempengaruhi minat konsumen untuk membeli. Faktor-faktor yang mempengaruhi peningkatan penjualan itu meliputi lokasi gerai, keanekaragaman produk, harga, promosi, desain dan tampilan gerai, pelayanan dan *personal selling* [4]. Hwang, dkk (2009) menyatakan bahwa tampilan inventori pada rak memiliki dampak pada penjualan di beberapa produk retail [5].

Larson dan DeMarais (1990) menyatakan bahwa stok psikis adalah inventori yang ditampilkan untuk meningkatkan penjualan dan menyarankan aturan '*full-shelf merchandising*' dimana area *display* selalu penuh dengan stok produk [6]. Saat ini semakin banyak perusahaan ritel yang beralih pada penataan produk dan pengalokasian rak *display* untuk memaksimalkan keuntungan [7]. Alokasi produk pada rak display merupakan permasalahan yang dihadapi oleh banyak perusahaan ritel. Masalah akan muncul ketika begitu banyak jumlah produk yang harus *display* dengan jumlah rak yang terbatas. Sehingga perlu dilakukan optimasi pengalokasian produk pada ruang rak *display*.

Beberapa pendekatan dan model telah dikembangkan untuk mencari solusi dalam peletakan produk dan pengalokasian rak. Hwang et al. (2009) mengembangkan model untuk menentukan pembagian ruang rak *display*, besarnya area rak, dan penempatan produk untuk setiap merek pada satu kategori produk [5]. Dalam pengembangannya, Hwang et al. (2009) menggunakan pendekatan algoritma genetika [5]. S Raut et al. (2009), mengembangkan model pengalokasian produk dengan mempertimbangkan periode-periode dimana produk *display* [8]. Nafari dan Shahrabi (2010) mengembangkan model pengalokasian produk berdasarkan hubungan antar kategori

dari produk dengan menambahkan variabel elastisitas harga menggunakan pendekatan *data mining* [7]. Castelli dan Vanneschi (2014) mengembangkan model S Raut dkk dengan menggunakan pendekatan algoritma genetika (*genetic algorithm*) dan pencarian variabel terdekat (*variable neighborhood search*) [9].

Berdasarkan penelitian-penelitian yang telah dilakukan sebelumnya, maka penelitian ini akan mengacu model pengalokasian produk pada ruang rak *display* yang dikembangkan oleh Hwang et al. (2009) dan Castelli (2014) dan Vanneschi dengan menggunakan metode algoritma genetika yang dimodifikasi dengan pemrograman integer. Sebuah model pengalokasian produk pada ruang rak *display* dapat memudahkan retailer minimarket untuk memaksimalkan keuntungan.

1.2 Perumusan Masalah

Tugas Akhir yang akan diajukan ini menitikberatkan permasalahan pada beberapa hal sebagai berikut :

1. Model pengalokasian produk pada rak *display* seperti apa yang sesuai dengan kondisi Minimarket Bentar?
2. Konfigurasi alokasi produk pada rak *display* seperti apa yang dapat meningkatkan keuntungan?

1.3 Batasan Tugas Akhir

Batasan-batasan dalam pembuatan Tugas Akhir ini sebagai berikut :

1. Sumber data yang digunakan berasal dari Minimarket Bentar cabang Delta Sari, Sidoarjo. Data tersebut mencakup data kapasitas ruang rak *display* dan data produk kategori susu bubuk anak yang terdistribusi di rak.
2. *Display* yang digunakan adalah tiga buah rak dengan lima *layer* berbentuk gondola dan tampak depan.
3. Kategori produk yang digunakan dalam penelitian adalah kategori produk susu bubuk anak.

1.4 Tujuan Tugas Akhir

Tujuan pembuatan tugas akhir ini adalah untuk memperoleh model dan konfigurasi alokasi produk pada ruang rak *display* di setiap layer rak sehingga dapat memaksimalkan jumlah *brand* produk yang di-*diplay*.

1.5 Manfaat Tugas Akhir

Manfaat yang akan diperoleh dengan adanya tugas akhir ini antara lain:

1. Bagi peritel, dapat mengetahui konfigurasi alokasi produk pada ruang rak *display* sehingga dapat mengoptimalkan keuntungan dari produk kategori susu bubuk anak di minimarket.
2. Bagi peneliti, dapat mengetahui kinerja algoritma genetika yang dimodifikasi apabila diterapkan untuk optimasi pengalokasian produk pada ruang rak *display* minimarket.

1.6 Sistematika Penulisan

Sistematika penulisan buku tugas akhir ini dibagi dalam enam bab sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini dijelaskan tentang latar belakang, perumusan masalah, batasan masalah tugas akhir, manfaat tugas akhir, tujuan tugas akhir dan sistematika penulisan tugas akhir.

BAB II TINJAUAN PUSTAKA

Pada bab ini akan dijelaskan tentang teori dan informasi yang berkaitan dengan tugas akhir. Penggalian teori dan informasi dilakukan melalui berbagai sumber seperti buku, jurnal, artikel ilmiah, dan artikel berita internet. Teori dan informasi yang digali pada tugas akhir ini antara lain mengenai manajemen ritel, manajemen ruang *display* dan algoritma genetika.

BAB III METODOLOGI Pengerjaan

Pada bab ini dijelaskan tentang langkah-langkah pengerjaan tugas akhir beserta metode yang dipakai, mulai dari studi literatur, mengumpulkan data, mendefinisikan model, eksekusi metode optimalisasi dengan algoritma genetika dan pemrograman integer, analisis hasil pengolahan data, menarik kesimpulan dan saran dan membuat buku tugas akhir.

BAB IV PERANCANGAN

Pada bab ini akan dijelaskan mengenai pendefinisian model matematis sesuai dengan kondisi yang ada di Minimarket Bentar. Model tersebut digunakan untuk menyelesaikan permasalahan dalam tugas akhir dengan bantuan aplikasi Matlab.

BAB V IMPLEMENTASI

Pada bab ini akan dijelaskan tahapan-tahapan dalam implementasi model matematis pada metode Algoritma Genetika dan Pemrograman Integer. Implementasi yang ada pada tahapan ini meliputi proses pengerjaan dengan menggunakan toolbox optimasi *ga* pada Matlab untuk algoritma genetika dan POM-QM untuk pemrograman integer.

BAB VI HASIL DAN PEMBAHASAN

Pada bab ini akan dipaparkan hasil implementasi model matematis pada metode Algoritma Genetika. Hasil optimal dari algoritma genetika akan diolah dengan pemrograman integer untuk mendapatkan hasil yang bulat. Hasil paling optimal lalu dibuat peta konfigurasi produk yang di-*display*.

BAB VII KESIMPULAN DAN SARAN

Pada bab ini akan dipaparkan kesimpulan dari pengerjaan tugas akhir dan saran untuk pengembangan lebih lanjut.

Halaman ini sengaja dikosongkan

BAB II

TINJAUAN PUSTAKA

Bab ini membahas teori dan konsep yang mendukung pengerjaan tugas akhir. Teori-teori yang digunakan adalah konsep alokasi produk pada bisnis ritel dan metode yang digunakan dalam melakukan optimasi alokasi produk.

2.1 Studi Sebelumnya

Beberapa penelitian telah dilakukan untuk mengembangkan model dan penerapan algoritma dalam mencari optimalisasi alokasi produk pada rak *display* di bisnis ritel. Berikut merupakan pemaparan singkat dari penelitian-penelitian yang pernah dilakukan.

Tabel 2. 1 Studi Sebelumnya

No	Tahun; Penulis	Pembahasan
1.	<i>A genetic algorithm approach to an integrated problem of shelf</i>	
	2009; Hark Hwang, Bum Choi, Grimi Lee	Pada penelitian ini, penulis menerapkan algoritma genetika untuk mengoptimalkan alokasi item pada rak <i>display</i> pada bisnis ritel. Penelitian ini memperhatikan macam-macam desain rak dalam alokasi item dan menggunakan dua tipe perhitungan untuk alokasi produk. Salah satu batasan dari penelitian ini adalah satu rak untuk satu merk produk, namun tidak semua minimarket menerapkan satu rak untuk satu merk. Hasil dari penelitian ini adalah model peletakan produk

		per kategori dalam konfiurasi ruang rak pada bisnis ritel.
2	<i>A temporal data mining approach for shelf-space allocation with consideration of product price</i>	
	2010; Maryam Nafari, Jamal Shahrabi	Tujuan dari penelitian ini adalah mengembangkan model pengalokasian rak dengan mempertimbangkan harga produk. Perubahan harga produk memungkinkan adanya perubahan asosiasi antar produk terutama produk komplementer dan produk pengganti. Perubahan harga produk tersebut nantinya dapat mempengaruhi strategi alokasi rak. Metode yang digunakan dalam penelitian ini adalah teknik penggalian data <i>association rules</i> dan algoritma apriori. Teknik <i>data mining</i> digunakan untuk menentukan hubungan antar produk sehingga dapat diketahui produk yang saling berkaitan dan memiliki keuntungan. Hasil dari penelitian ini adalah konfigurasi peletakan produk yang mempertimbangkan aspek harga.
3	<i>Genetic algorithm with variable neighborhood search for optimal allocation of goods in shop shelves</i>	
	2014; Mauro Castelli, Leonardo Vanneschi	Tujuan dari studi ini adalah mengembangkan gabungan dua algoritma yaitu algoritma genetika dan pencarian variabel terdekat (<i>variable neighborhood</i>

		<p><i>search</i>) untuk mengoptimalkan alokasi produk di rak toko dan meningkatkan keuntungan peritel dimana pada penelitian ini algoritma pencarian variabel terdekat membantu mempersingkat proses eksekusi algoritma genetika. Hasil dari penelitian ini adalah penerapan algoritma GA-VNS pada permasalahan alokasi produk di rak toko dan menganalisis kecocokan metode dengan permasalahan.</p>
4	<p><i>Optimal Retail Shelf Space Allocation with Dynamic Programming using Bounds</i></p>	
	<p>2008; Hasmukh K.Gajjar, Gajendra K. Adil</p>	<p>Pada penelitian ini penulis menerapkan model <i>non-linear shelf-space allocation model</i> (NLSSAM) dengan <i>new dynamic programming</i> (NDP) untuk mengoptimalkan alokasi item pada rak <i>display</i> pada bisnis ritel. Hasil pada penelitian ini membuktikan bahwa NDP dengan menggunakan batasan lebih efisien dalam menyelesaikan permasalahan besar daripada <i>originial dynamic programming</i> (ODP) tanpa menggunakan batasan.</p>
5	<p><i>Heuristic and meta-heuristic approaches for multi-period shelf-space optimization: the case of motion picture retailing</i></p>	
	<p>2009; S Raut, S Swami, MP Moholkar</p>	<p>Studi ini mengembangkan model matematis optimasi alokasi ruang rak yang telah ada.</p>

		<p>Pada studi ini, penulis mengembangkan ke permasalahan pengalokasian produk multi-periode pada bisnis perfilman. Penelitian ini memformulasikan model <i>linear integer programming</i> untuk merepresentasikan permasalahan. Selain itu studi ini menggunakan dua algoritma yaitu algoritma <i>greedy</i> dan algoritma genetika. Hasil dari studi ini menunjukkan bahwa algoritma genetika memberikan hasil yang lebih baik daripada metode yang pernah ada.</p>
--	--	--

2.2 Dasar Teori

2.2.1 Minimarket Bentar

Minimarket Bentar merupakan sebuah jaringan minimarket yang menyediakan kebutuhan pokok dan kebutuhan sehari-hari. Minimarket ini didirikan pada tahun 1985 yang bertempat di kawasan perumahan Delta Sari, Sidoarjo dan dikelola oleh pribadi. Minimarket Bentar diawali sebagai bisnis ritel tradisional yang menjual beraneka ragam kebutuhan masyarakat sekitar, seperti sayur, buah, bumbu dapur, dan keperluan sehari-hari. Banyaknya kompetitor yang datang memaksa Minimarket Bentar untuk mengubah konsep ritel tradisional menjadi ritel modern untuk menarik pelanggan. Hingga kini, Minimarket Bentar memiliki 10 gerai yang tersebar di Jawa Timur.

Dalam melakukan *display* produk, pramuniaga Minimarket Bentar meletakkan barang-barang sesuai dengan rak-rak yang ditentukan berdasarkan kategori barang makanan atau bukan makanan, ukuran barang, dan merek barangnya. Minimarket Bentar juga memiliki strategi meletakkan barang

substitusi (pengganti) jika ada rak yang kosong dan produk sedang mengalami *out of stock*. Strategi ini dilakukan dengan meletakkan barang-brang lain yang sejenis untuk mengisi rak yang kosong dan mengganti label harga pada barang yang kosong dengan tulisan persediaan habis, agar konsumen tidak salah melihat harga.

Rak *display* yang digunakan Minimarket Bentar memiliki berbagai macam bentuk dan menyesuaikan dengan kondisi ruangan. Rak yang digunakan untuk ruangan yang kecil memiliki tinggi 150 cm, panjang 90 cm, dan lebar 40 cm. Sedangkan untuk ruangan yang lebih besar menggunakan rak yang memiliki tinggi 180 cm, panjang 90 cm, dan lebar 40 cm. Rak tersebut dinamakan dengan *island gondola* karena diletakkan di tengah area toko. Jarak antara rak satu dengan rak lainnya di lorong adalah 50 cm.



Gambar 2. 1 Rak Berbentuk *Island Gondola*

Kapasitas yang ditampung rak bergantung kepada ukuran produk yang akan *didisplay*. Rak pada tingkatan paling atas menampung produk dengan ukuran yang lebih kecil, begitu pula sebaliknya, rak pada tingkatan paling bawah menampung produk dengan ukuran yang lebih besar. Rak paling rendah menampung kurang lebih 16 produk berukuran besar. Rak

paling atas menampung kurang lebih 40 produk berukuran kecil.

2.2.2 *Retailing*

Kata “ritel” mengandung arti eceran atau perdagangan eceran, dan peritel (*retailer*) dapat diartikan sebagai pengecer atau pengusaha perdagangan eceran. *Retailing* merupakan sekumpulan aktivitas bisnis yang menambahkan nilai pada produk dan jasa untuk dijual kepada pelanggan untuk digunakan pribadi atau keluarga [10]. Menurut Berman dan Evans (1998), *Retailing* terdiri dari aktivitas bisnis yang melibatkan penjualan barang dan jasa kepada konsumen untuk kebutuhan pribadi, keluarga atau rumah tangga [11]. *Retailing* merupakan segala sesuatu yang mencakup kegiatan penjualan barang atau jasa kepada konsumen akhir untuk digunakan pribadi, keluarga, atau rumah tangga, bukan untuk penggunaan bisnis.

Dalam rantai pasok, *retail* merupakan bagian terakhir yang menghubungkan manufaktur ke pelanggan. Peran peritel adalah menyalurkan produk yang ada pada distributor (*Wholesaler*) ke pelanggan akhir. Fungsi ini dapat dilihat dari Gambar 2.2 yang menggambarkan proses sederhana pendistribusian produk dari manufaktur kepada pelanggan.



Gambar 2. 2 Proses *Supply Chain*

2.2.2.1 Fungsi Ritel

Perusahaan membutuhkan jasa *retailer* dalam jaringan rantai pasok karena *retailer* dapat menciptakan nilai tambah dari barang atau jasa yang ditawarkan oleh perusahaan. Selain itu, *retailer* memfasilitasi perusahaan di bagian distribusi barang atau jasa dari pabrik atau *wholesaler* ke konsumen akhir. Menurut Levy dan Weitz (2009), nilai tambah yang diciptakan *retailer* [10] meliputi.

1. Menyediakan berbagai macam jenis barang dan jasa
Retailer memiliki keahlian dalam menspesialisasi keanekaragaman produk yang ditawarkan. Sehingga, konsumen dapat dengan mudah mencari bermacam-macam jenis barang di toko *retail*. Keanekaragaman jenis barang dan jasa dapat meningkatkan keinginan konsumen untuk berbelanja di toko tersebut.
2. Menjual dalam jumlah yang sedikit
Untuk mengurangi biaya transportasi, manufaktur atau *wholesaler* mengirim barang ke *retailer* dalam jumlah yang besar. Selanjutnya, *retailer* akan membaginya dalam jumlah lebih sedikit untuk dijual kepada konsumen. Konsumen dapat membeli barang dalam jumlah yang lebih sedikit sesuai dengan kebutuhannya dan perusahaan dapat mengefisiensikan biaya transportasi dengan mengirimkan barang dalam jumlah yang besar.
3. Menyimpan *inventory*
Salah satu fungsi utama dari *retailer* adalah menyimpan produk dengan kuantitas yang cukup sehingga produk tersedia ketika konsumen membutuhkannya. Konsumen dapat menyimpan produk dengan kemasan yang kecil karena mengetahui bahwa *retailer* memiliki stok produk tersebut.
4. Menyediakan jasa penjualan
Retailer menyediakan jasa penjualan yang memudahkan pelanggan dalam membeli dan menggunakan produk, seperti penggunaan kartu kredit dalam pembelian produk. *Retailer* juga menampilkan produk, sehingga pelanggan dapat melihat dan mencoba produk tersebut sebelum dibeli.

2.2.2.2 Jenis Bisnis Ritel

Bisnis ritel dapat diklasifikasikan berdasarkan kepemilikan bisnis, barang dagang, luas area penjualan, dan *retailer* tanpa toko [12].

2.2.2.2.1 Bisnis Ritel Berdasarkan Kepemilikan

- *Single-store Retailer* merupakan bisnis ritel yang memiliki jumlah paling banyak. Bisnis ini memiliki ukuran toko yang umumnya di bawah 100 m² dan dapat berbentuk kios, toko di pasar tradisional serta *minimarket* modern dengan kepemilikan secara individual.
- Rantai Toko Ritel merupakan bisnis ritel yang memiliki lebih dari satu cabang dengan kepemilikan dalam bentuk perseroan. Bentuk dari bisnis ini dimulai dari *minimarket* hingga *hyperstore*.
- Toko Waralaba (*Franchise Store*) merupakan bisnis ritel berdasarkan kontrak kerja dan bagi hasil antara penyewa waralaba (*franchisee*) dengan pemilik waralaba (*franchisor*).

2.2.2.2.2 Bisnis Ritel Berdasarkan Barang Dagang

- *Specialty Store* merupakan bisnis ritel yang hanya menjual satu jenis kategori barang atau beberapa kategori barang yang dijualnya relatif sedikit.
- *Grocery Store* merupakan bisnis ritel yang sebagian besar menyediakan kategori barang kebutuhan sehari-hari, makanan segar, barang pecah belah, makanan ringan, susu bubuk, kosmetik, dan peralatan rumah tangga.
- *Department Store* merupakan bisnis ritel dengan kategori barang yang dijual merupakan barang kebutuhan non-pokok, seperti pakaian.
- *Hyperstore* merupakan bisnis ritel menjual hampir seluruh jenis barang kebutuhan untuk

setiap segmen konsumen dengan konsep *one stop shopping*. Umumnya luas area untuk bisnis ritel ini sedikitnya 10.000 m².

2.2.2.2.3 Bisnis Ritel Berdasarkan Luas Area Penjualan

- *Small Store* merupakan sebuah toko kecil yang umumnya adalah toko tradisional dengan luas area penjualan kurang dari 100 m².
- *Minimarket* merupakan toko yang beroperasi dengan luas area penjualan antara 100 hingga 1.000 m².
- *Supermarket* merupakan toko yang dioperasikan dengan luas area penjualan antara 1.000 hingga 5.000 m².
- *Hypermarket* merupakan toko yang beroperasi dengan luas area penjualan diatas 5.000 m².

2.2.2.2.4 Bisnis Ritel Tanpa Toko

- *Multi-Level-Marketing* (MLM) merupakan suatu model penjualan barang secara langsung sesuai dengan sistem komisi penjualan berperingkat yang didasarkan oleh keanggotaan dalam jalur distribusi.
- *Mail & Phone Order Retailer*, merupakan perusahaan yang melakukan penjualan berdasarkan pesanan melalui surat atau telepon.
- *Online-Store* (*e-commerce*) merupakan perusahaan yang melakukan penjualan melalui internet.

2.2.3 Manajemen Ruang *Display*

Manajemen ruang (*space management*) merupakan suatu konsep atau rencana pemajangan produk berdasarkan alur kebiasaan belanja konsumen dengan tujuan untuk memaksimalkan keuntungan dan meningkatkan pelayanan kepada konsumen akhir [13]. Manajemen ruang *dispay* diperlukan untuk mengoptimalkan pengalokasian

keanekaragaman produk pada ruang rak *display* yang terbatas. Menurut Levy dan Weitz [10], manajemen ruang melibatkan dua sumber keputusan yaitu alokasi ruang toko untuk menampilkan kategori dan merek dan lokasi dari kategori barang di toko.

Penerapan manajemen ruang memerlukan beberapa pertimbangan yaitu sebagai berikut.

- Barang yang ada didisplay dicocokkan dengan permintaan pelanggan sehingga dapat dilakukan perencanaan yang tepat untuk stok yang tersedia untuk dibeli oleh pelanggan.
- Menyediakan perkiraan jumlah produk untuk ditampilkan di rak secara tepat.
- Menyusun produk berdasarkan kategori dan kinerja produk dalam gerai.

Manajemen ruang *display* merupakan aspek penting pada bisnis ritel karena berdampak pada kebiasaan pelanggan dalam berbelanja. Pada manajemen ruang *display*, produk ditata semenarik mungkin sehingga dapat menarik pelanggan dan produk dikelompokkan sesuai dengan jenis dan fungsinya. Perbedaan antara rak *display* gerai dengan rak yang ada di gudang adalah rak *display* pada minimarket barang harus selalu tersedia dan menampilkan informasi harga mengenai produk sedangkan penataan pada rak gudang dapat ditumpuk dan disesuaikan dengan kapasitas gudang.

2.2.4 Algoritma Genetika

Algoritma Genetika merupakan salah satu teknik pencarian global yang digunakan untuk mencari solusi optimal dari permasalahan optimasi yang kompleks. Algoritma ini menunjukkan diri dengan dan beradaptasi sama seperti organisme biologis yaitu dengan mengikuti proses evolusi makhluk [14]. Metode yang digunakan untuk meningkatkan solusi adalah dengan memproduksi keturunan dengan menggunakan solusi terbaik dari generasi saat ini yang disebut sebagai “*parent*”.

2.2.4.1 Definisi Penting pada Algoritma Genetika

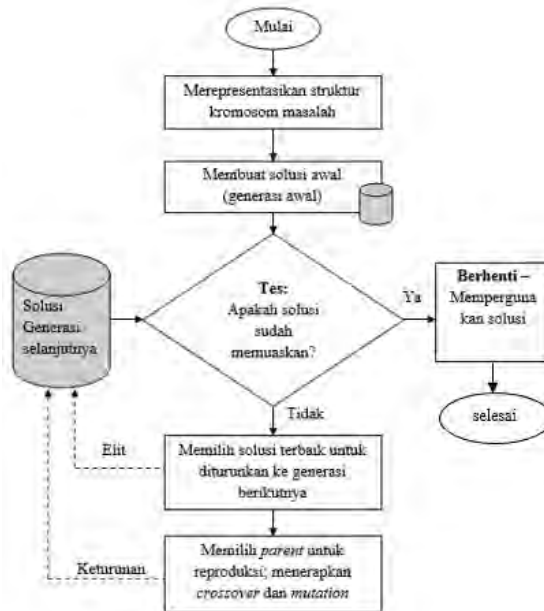
Terdapat beberapa istilah yang digunakan pada algoritma genetika yang juga merupakan istilah biologis. Istilah-istilah yang digunakan adalah sebagai berikut.

- *Chromosome*, merupakan gabungan dari gen-gen yang membentuk nilai tertentu dan menyatakan kemungkinan solusi dari suatu permasalahan.
- *Gen*, merupakan nilai yang menyatakan satuan dasar yang membentuk suatu arti tertentu dalam satu kesatuan gen yang disebut kromosom.
- *Locus*, merupakan posisi dari gen.
- *Allele*, merupakan nilai yang dimasukkan dalam gen.
- *Penotype*, merupakan kromosom yang berperan sebagai solusi akhir.
- *Genotype*, merupakan calon-calon solusi.
- *Population*, merupakan kumpulan dari kromosom/calon solusi.
- *Reproduction*, merupakan sebuah proses untuk mendapatkan generasi potensial dengan memilih *parent* dengan nilai lebih tinggi atau dengan memilih *parent* yang mempunyai kemungkinan besar untuk dipilih saat proses reproduksi.
- *Crossover*, memilih posisi acak dalam deretan dan bertukar segmen baik ke kanan atau ke kiri dengan segmen lain di dalam deretan untuk menghasilkan dua keturunan baru.
- *Mutation*, merupakan perubahan acak dalam representasi kromosom.
- *Elitism*, merupakan proses mempertahankan keturunan terbaik untuk berkembang melalui generasi.

2.2.4.2 Proses Algoritma Genetika

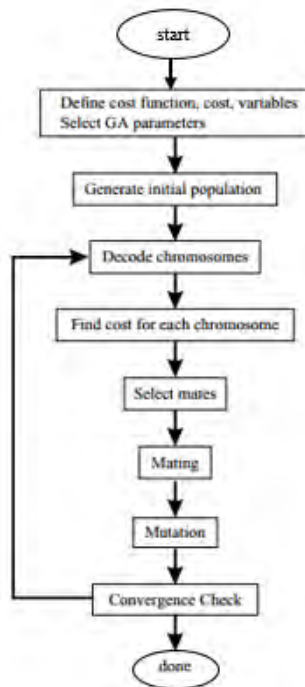
Gambar 2.3 menunjukkan *flow diagram* dari proses algoritma genetika. Masalah yang akan diselesaikan harus mendeskripsikan dan merepresentasikan algoritma genetika yang dapat disetujui. Algoritma genetika menggunakan

pemilihan *parent* secara acak sehingga dapat menghasilkan keturunan dengan melalui operator genetik (contoh: *crossover*, *mutation*). Jumlah solusi potensial ditentukan oleh besarnya populasi yang ditentukan parameter acak sebelum sebelum solusi relovusi. Ketika generasi baru telah terbentuk, maka solusi tersebut akan melalui evaluasi dan terbentuk populasi baru untuk beberapa iterasi.



Gambar 2. 3 Flow Diagram Algoritma Genetika (Sumber: Turban, 2010)

Haupt (2004), membedakan algoritma genetika menjadi dua yaitu Algoritma Genetika Biner dan Algoritma Genetika Kontinu [15]. Pada algoritma genetika biner, angka biner merepresentasikan sebagai karakteristik dari variabel yang digunakan sedangkan pada algoritma genetik kontinu, variabel bersifat kontinu dan memiliki jumlah yang banyak sehingga harus digunakan angka desimal untuk mencari solusi. Gambar 2.4 menunjukkan *flow diagram* pada algoritma genetika kontinu.



Gambar 2. 4 Flowchart Algoritma Genetika Kontinu (Sumber: Haupt, 2004)

2.2.5 Integer Programming

Integer Programming merupakan salah satu pemodelan untuk pencarian solusi yang optimal dari suatu permasalahan. *Integer Programming* digunakan untuk memodelkan suatu permasalahan yang variabel-variabelnya berupa bilangan bulat, seperti jumlah orang dan jumlah barang yang tidak mungkin berupa pecahan.

Menurut Taylor (2007), terdapat tiga tipe dasar dari model pemrograman integer [16] yaitu sebagai berikut.

- Model Integer Keseluruhan.
Pada model ini, semua variabel keputusan memiliki nilai berupa bilangan bulat. Model

matematika untuk model integer keseluruhan sebagai berikut.

$$\text{Maksimum : } Z = 100x_1 + 150x_2$$

$$\text{Batasan : } 8,000x_1 + 4,000x_2 \leq 40,000$$

$$15x_1 + 30x_2 \leq 200$$

$$x_1, x_2 \geq 0 \text{ dan integer}$$

- Model Integer 0-1.

Pada model ini, nilai solusi pada variabel keputusan adalah nol atau satu. Model matematika untuk model integer 0-1 adalah sebagai berikut.

$$\text{Maksimum : } Z = 300x_1 + 150x_2 + 200x_3$$

$$\text{Batasan : } 80x_1 + 40x_2 + 10x_3 \leq 400$$

$$15x_1 + 30x_2 + 20x_3 \leq 200$$

$$x_1, x_2, x_3 = 0 \text{ atau } 1$$

- Model Integer Campuran.

Model ini merupakan model pemrograman integer yang hasil dari variabel keputusan bernilai bilangan bulat atau bilangan non-bulat. Model matematika untuk model integer campuran adalah sebagai berikut.

$$\text{Maksimum : } Z = 9,000x_1 + 1,500x_2 + 1,000x_3$$

$$\text{Batasan : } 50,000x_1 + 12,000x_2 + 8,000x_3 \leq 250,000$$

$$x_1 \leq 4$$

$$x_2 \leq 15$$

$$x_3 \leq 20$$

$$x_2 \geq 0$$

$$x_1, x_3 \geq 0 \text{ dan integer}$$

2.2.5.1 Algoritma Branch and Bound

Algoritma Branch and Bound adalah metode algoritma umum untuk mencari solusi optimal dari berbagai permasalahan optimasi, terutama untuk optimasi diskrit dan kombinatorial [17]. Ruang solusi algoritma *branch and bound* diorganisasikan ke dalam pohon ruang status dimana permasalahan dibagi-bagi menjadi subregion-subregion yang mungkin mengarah ke solusi. Langkah utama dari algoritma

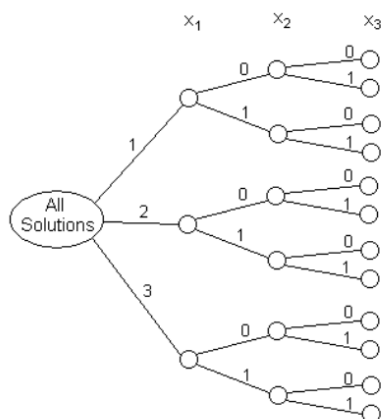
ini adalah pembatasan (*bound*), pencabangan cabang (*branching*) dan penghentian cabang (*fathoming*).

Pada proses *branching*, pencabangan dilakukan jika masih terdapat variabel keputusan yang harus bernilai integer namun memiliki solusi yang tidak integer. Pencabangan dilakukan dengan cara menambahkan batasan pada permasalahan awal. Penambahan batasan ini ditujukan untuk membuat variabel keputusan yang belum bernilai integer supaya bernilai integer.

Proses *bounding* terdapat dua batasan yaitu batas atas (*upper bound*) dan batas bawah (*lower bound*). Pada permasalahan maksimisasi, batas atas merupakan solusi dari pemograman integer sedangkan batas bawahnya adalah nilai dari sub permasalahan yang dihadapi atau sub permasalahan lain yang semua variabel keputusannya harus bernilai integer.

Proses *Fathoming* atau penghentian pencabangan dilakukan apabila *infeasible* atau tidak mempunyai daerah layak, semua variabel keputusan bernilai integer, dan kondisi menunjukkan nilai optimal.

Struktur algoritma branch and bound menyerupai bentuk pohon seperti pada Gambar 2.5 yang berbaring pada sisinya dengan *root* pada kiri, berlabel “*all solution*” dan daun di sebelah kanan. Titik daun merupakan solusi enumerasi yang sebenarnya sehingga terdapat 12 solusi: (3 kemungkinan nilai x_1) \times (2 kemungkinan nilai x_2) \times (2 kemungkinan nilai x_3). Sebagai contoh, titik utama (*root node*) mempresentasikan semua solusi yang dapat dihasilkan dengan menumbuhkan pohon. Titik tengah (*intermediate node*) lainnya, contohnya titik pertama langsung ke kanan dari titik utama (*root node*), merupakan bagian lain semua solusi yang mungkin, dalam hal ini, semua solusi dimana $x_1 = 2$ dan dua variabel lainnya dapat mengambil salah satu dari nilai-nilai yang mungkin. Untuk dua titik yang terhubung langsung di pohon, simpul orangtua (*parent node*) adalah yang paling dekat dengan akar dan simpul anak (*child node*) adalah yang paling dekat dengan daun.

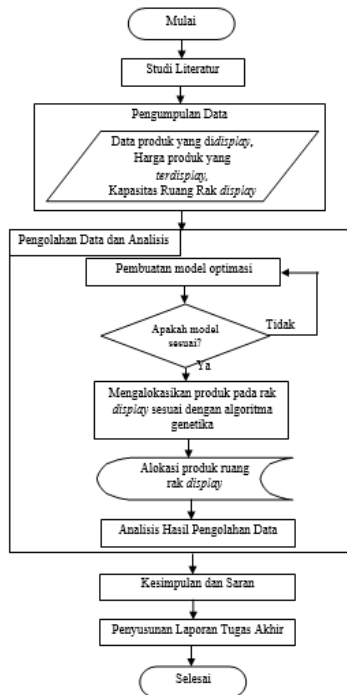


Gambar 2. 5 Pohon Enumerasi pada Branch and Bound

BAB III

METODOLOGI PENELITIAN

Bab ini membahas tentang metodologi yang digunakan dalam pengerjaan tugas akhir. Metodologi ini digunakan sebagai panduan agar pengerjaan tugas akhir berjalan terarah dan sistematis. Metodologi pelaksanaan Tugas Akhir ini dibagi menjadi beberapa tahapan antara lain studi literatur, pengumpulan data, pembuatan model, analisis hasil, dan penarikan kesimpulan. Pada bab ini akan dijelaskan secara detil alur pengerjaan yang telah disebutkan di atas. Alur dan urutan metodologi ditunjukkan oleh Gambar 3.1.



Gambar 3. 1 Metodologi Pengerjaan Tugas Akhir

3.1 Studi literatur

Pada tahap ini akan dilakukan pendalaman lebih lanjut mengenai landasan teori yang akan digunakan dalam pengerjaan tugas akhir terkait materi algoritma genetika, pemrograman integer, alokasi produk pada bisnis *retail*, dan optimasi. Referensi materi diperoleh dari beberapa sumber sebagai berikut.

- Buku. Buku-buku yang digunakan adalah buku tentang bisnis *retail* yang nanti akan ditemukan referensi mengenai manajemen ruang (*space management*) yang termasuk manajemen rak *display*. Buku ini dicari di perpustakaan ITS, perpustakaan Universitas Airlangga maupun di toko buku terdekat.
- E-book. E-book mengenai manajemen *retail*, dan algoritma genetika. Kebanyakan dari buku elektronik ini nanti akan didapatkan melalui fasilitas books.google.com.
- Jurnal. Jurnal yang digunakan referensi untuk mengetahui tentang penelitian terdahulu terkait dengan dengan algoritma genetika dan alokasi *display* rak pada bisnis ritel. Pada pencarian jurnal ini menggunakan kata kunci *genetic algorithm*, *space allocation*, *retail*, *shelf*, dan *display*. Pencarian jurnal ini melalui sebuah website resmi [sciencedirect.com](https://www.sciencedirect.com), [IEEE.com](https://www.ieee.com), dan search.proquest.com
- Artikel. Artikel ini digunakan sebagai referensi untuk mencari perkembangan bisnis ritel saat ini di Indonesia serta pentingnya penerapan manajemen ritel pada bisnis ritel. Artikel ini lebih mengkhususkan tentang artikel berita dari beberapa sumber seperti [Liputan6.com](https://liputan6.com), swa.co.id, kompas.co.id, dan detik.com.

Berdasarkan pencarian studi literatur tersebut akan diperoleh dugaan-dugaan awal yang akan berkontribusi dalam pengerjaan tugas akhir ini.

3.2 Pengumpulan data

Pada tahap ini, data yang dikumpulkan berkaitan dengan alokasi produk yang *terdisplay* di minimarket Bentar. Data-data yang akan digunakan dalam pengerjaan tugas akhir ini diantaranya adalah data item produk, harga produk, kuantitas produk, ukuran rak, dan kapasitas rak. Pengumpulan data ini dilakukan dengan wawancara oleh pengelola minimarket Bentar cabang Delta Sari Sidoarjo

3.3 Pengolahan Data dan Analisis

3.3.1 Pendefinisian Model Optimasi

Pada tahap ini akan dilakukan pendefinisian model optimasi yang sesuai dengan permasalahan. Model optimasi yang akan didefinisikan adalah model untuk memaksimalkan keuntungan. Model optimasi yang digunakan untuk model memaksimalkan keuntungan mengacu pada model Castelli [9] dengan fungsi tujuan memaksimalkan keuntungan peritel. Fungsi tujuan yang diusulkan Castelli adalah sebagai berikut.

$$\text{Maximize } \sum_{i=1}^n \sum_{k=1}^m p_{ik} x_{ik} \quad (3.1)$$

Variabel :

p_{ik} Profit produk i pada rak k

x_{ik} Jumlah produk i pada rak k

Berikut merupakan beberapa asumsi yang diberikan dalam pengerjaan :

- Ukuran panjang, lebar, dan tinggi satu rak yang digunakan adalah 90 cm, 40 cm, dan 150 cm, terdapat lima *layer*.
- Jumlah merek produk *ter-display* pada rak adalah 79 produk.
- Batasan lebar produk diabaikan.
- Rata-rata ukuran kemasan setiap produk dibedakan menjadi tiga jenis yaitu sebagai berikut:

- Ukuran kecil (untuk produk yang memiliki berat 150 g hingga 200 g) = 10 cm x 3,5 cm x 16 cm,
- sedang (untuk produk yang memiliki berat 350 g hingga 400 g) = 15 cm x 5 cm x 20 cm,
- besar (untuk produk yang memiliki berat 700 g hingga 1 kg) = 16 cm x 6 cm x 25 cm.
- Dalam satu rak terdapat lima *layer* dengan tinggi yang berbeda-beda, yaitu sebagai gambar berikut.

Layer 1	} ~ cm
Layer 2	} 30 cm
Layer 3	} 30 cm
Layer 4	} 40 cm
Layer 5	} 40 cm

Gambar 3. 2 Ilustrasi Rak Displai

- Kapasitas pada setiap layer berbeda-beda tergantung produk yang didisplai, namun panjang pada setiap layer sama yaitu 270 cm.
- Produk kategori susu bubuk anak yang paling laris pada minimarket Bentar adalah Milo (dalam berbagai ukuran), Bebelove 3, Bebelove 4, dan Indomilk Coklat.
- Penataan dilakukan pada satu sisi, yaitu tampak depan.
- Harga produk diketahui dan konstan.
- Permintaan produk dipengaruhi oleh letak dan kuantitas item yang ter-*display*.
- Satu *brand* produk hanya menempati satu ruang.

3.3.2 Pengalokasian Produk pada Rak

Pada tahap ini dilakukan pengalokasian produk sesuai model yang telah dibuat. Prosedur pengalokasian produk yang dilakukan pada penelitian ini mengacu pada model yang dikembangkan oleh Castelli (2014) yang kemudian disesuaikan dengan minimarket Bentar. Pada tahap ini dilakukan proses optimasi menggunakan Algoritma Genetika. Proses kerja algoritma genetika dapat direpresentasikan pada diagram alir seperti Gambar 2.4. Pada proses optimasi ini nantinya model memaksimalkan keuntungan atau memaksimalkan jumlah *brand* produk yang di-*display* akan dikerjakan secara bergantian.

3.3.3 Analisis Hasil Pengolahan Data

Pada tahap ini akan dilakukan analisis terhadap model yang telah diterapkan dengan Algoritma Genetika yang dimodifikasi. Hasil analisis tersebut akan dipresentasikan dalam bentuk peta konfigurasi peletakan produk pada rak *display* minimarket. Sehingga berdasarkan representasi tersebut dapat dilihat pengaruh peletakan produk terhadap keuntungan yang diperoleh peritel.

3.4 Pembuatan Kesimpulan dan Saran

Setelah dilakukan proses analisis terhadap data dan model optimasi, maka akan menghasilkan sebuah konfigurasi alokasi produk di rak *display* dan hasil dari model optimasi. Konfigurasi alokasi tersebut digunakan untuk menarik kesimpulan sehingga dapat memberikan solusi atau usulan kepada perusahaan dalam peningkatan kinerja perusahaan baik internal maupun eksternal. Berdasarkan solusi yang ditarik pada bagian kesimpulan, terdapat kelebihan dan kekurangan yang nantinya akan dilaporkan ke perusahaan sebagai saran atau masukan dari hasil pengerjaan tugas akhir ini.

3.5 Penyusunan Laporan Tugas Akhir

Tahapan selanjutnya adalah penyusunan laporan tugas akhir dalam bentuk buku Tugas Akhir. Pada penyusunan buku Tugas Akhir ini disertakan pendokumentasian proses-proses yang dilakukan oleh penulis selama melakukan survei ke Minimarket Bentar.

BAB IV

PERANCANGAN

Bab ini menjelaskan mengenai perancangan algoritma genetika yang dimodifikasi dengan pemrograman integer yang diimplementasikan. Perancangan yang dilakukan terdiri atas pengolahan data, pembuatan model optimasi, dan perancangan algoritma.

4.1 Pengolahan Data

Data yang dirancang dalam tahap ini meliputi data masukan, data selama proses, dan data keluaran yang dihasilkan oleh perangkat lunak.

4.1.1 Data Masukan

Data masukan yang digunakan dalam penelitian berupa file .xls yang berisi informasi mengenai data produk kategori susu bubuk anak. Informasi yang terdapat dalam file tersebut antara lain adalah nama item, ukuran per item (panjang, lebar, tinggi), harga per item, dan keuntungan per item. Keuntungan per item didapatkan 5% harga jual per item.

4.1.2 Data Proses

Data proses merupakan data masukan yang diolah menjadi beberapa persamaan yang nantinya akan digunakan sebagai fungsi fitness. Data proses merupakan data masukan yang telah diimpor ke dalam toolbox MATLAB untuk diolah menjadi data keluaran.

4.1.3 Data Keluaran

Data keluaran berupa tabel yang menunjukkan kuantitas produk yang didisplay sesuai dengan hasil yang dikeluarkan

oleh MATLAB. Data keluaran digunakan untuk menentukan konfigurasi alokasi produk yang sesuai dengan kebutuhan perusahaan.

4.2 Pembuatan Model Optimasi

Pembuatan model optimasi didasarkan pada data yang diperoleh, yaitu meliputi data ukuran per item, data profit per item, dan data rak yang digunakan dalam mendisplay produk kategori susu bubuk pada minimarket. Model optimasi digunakan mengacu pada model optimasi Castelli [9], seperti yang telah dituliskan pada Persamaan (3.1). Dalam pengerjaan tugas akhir ini, terdapat 5 layer pada rak yang digunakan dimana masing-masing memiliki model yang berbeda. Sehingga nantinya model akan dibahas layer per layer. Pembuatan model optimasi dibagi menjadi tiga bagian yaitu pendefinisian variabel keputusan, formulasi fungsi tujuan dan fungsi batasan.

4.2.1 Pendefinisian Variabel Keputusan

Pada bagian ini dilakukan pendefinisian variabel keputusan pada setiap layer dimana pendefinisian ini dibagi menjadi lima bagian. Satu bagian menunjukkan produk apa saja yang perlu didisplay pada satu layer. Berikut merupakan variabel keputusan yang digunakan dalam pengerjaan tugas akhir berdasarkan layer per layer.

• Layer 1

- x_1 : kuantitas produk Bebelove 1
- x_2 : kuantitas produk Bebelove 2
- x_3 : kuantitas produk Milo 150gr
- x_4 : kuantitas produk Milo 300gr
- x_5 : kuantitas produk Milo 3 in 1 150gr
- x_6 : kuantitas produk Morinaga BMT 200gr
- x_7 : Kuantitas produk Morinaga Chil Mil 200 gr
- x_8 : kuantitas produk Morinaga Chil-Kid Madu 200gr

- X₉ : kuantitas produk Chil-Kid Vanilla 200gr
- X₁₀ : kuantitas produk Chil-School Coklat 200gr
- X₁₁ : kuantitas produk Chil-School Stroberi 200gr
- X₁₂ : kuantitas produk Chil-School Vanilla 200gr
- X₁₃ : kuantitas produk Ovaltine 150gr
- X₁₄ : kuantitas produk Ovaltine 280gr
- X₁₅ : kuantitas produk Ovaltine 3 in 150gr
- X₁₆ : kuantitas produk Vitalac Madu 150gr
- X₁₇ : kuantitas produk Vitalac Vanilla 150gr

- **Layer 2**

- X₁ : kuantitas produk Bebelove 3
- X₂ : kuantitas produk Bebelove 4
- X₃ : kuantitas produk Lactona 1 200gr
- X₄ : kuantitas produk Lactona 1 400gr
- X₅ : kuantitas produk Milo 600gr
- X₆ : kuantitas produk Milo 3 in 1 600gr
- X₇ : Kuantitas produk Morinaga BMT 400gr
- X₈ : kuantitas produk Morinaga Chil-Mid 400gr
- X₉ : kuantitas produk Chil-Kid Madu 400gr
- X₁₀ : kuantitas produk Chil-School Regular 400gr
- X₁₁ : kuantitas produk Ovaltine 600gr
- X₁₂ : kuantitas produk Ovaltine 3 in 1 600gr
- X₁₃ : kuantitas produk Vitalac Madu 400gr
- X₁₄ : kuantitas produk Vitalac Vanilla 400gr
- X₁₅ : kuantitas produk Vitalac Vanilla 800gr

- **Layer 3**

- X₁ : kuantitas produk Lactona 2 200gr
- X₂ : kuantitas produk Lactona 2 400gr

- x_3 : kuantitas produk Morinaga Chil-School Regular 800gr
- x_4 : kuantitas produk Morinaga Chil-School Platinum 800gr
- x_5 : kuantitas produk Zee Regular Chocholate Twist
- x_6 : kuantitas produk Zee Regular Strawberry Milk
- x_7 : kuantitas produk Zee Platinum Vanilla Delight
- x_8 : kuantitas produk Zee Platinum Choco Rich
- x_9 : kuantitas produk Zee Kidz 5+ Vanilla
- x_{10} : kuantitas produk Zee Kidz 5+ Honey
- x_{11} : kuantitas produk Zee Kidz 5+ Chocolate Milk
- x_{12} : kuantitas produk Zee Kidz Platinum Vanilla Delight
- x_{13} : kuantitas produk Zee Tween Chocolate Milk
- x_{14} : kuantitas produk Zee Tween Strawberry
- x_{15} : kuantitas produk Zee Tween Platinum Choco
- x_{16} : kuantitas produk Zee Tween Platinum Strawberry Ice

• **Layer 4**

- x_1 : kuantitas produk Bimbi 1 150gr
- x_2 : kuantitas produk Bimbi 1 400gr
- x_3 : kuantitas produk Bimbi Lola 130gr
- x_4 : kuantitas produk Boneto Coklat 700gr
- x_5 : kuantitas produk Boneto Coklat 350gr
- x_6 : kuantitas produk Boneto Coklat 150gr
- x_7 : kuantitas produk Indomilk Biokids 1-3 tahun 400gr
- x_8 : kuantitas produk Indomilk Biokids 4-6 tahun Coklat 400gr

- x_9 : kuantitas produk Indomilk Biokids 4-6 tahun Plain 400gr
- x_{10} : kuantitas produk Indomilk Coklat 400gr
- x_{11} : kuantitas produk Indomilk Plain 400gr
- x_{12} : kuantitas produk Lactona 3 Madu
- x_{13} : kuantitas produk Lactona 3 Vanilla
- x_{14} : kuantitas produk S-26 (1) 400gr
- x_{15} : kuantitas produk S-26 Nutrisure Gold 400gr
- x_{16} : kuantitas produk S-26 Promil 400gr

• **Layer 5**

- x_1 : kuantitas produk Bimbi 2 400gr
- x_2 : kuantitas produk Bimbi 3 400gr
- x_3 : kuantitas produk Boneto Madu 700gr
- x_4 : kuantitas produk Boneto Madu 350gr
- x_5 : kuantitas produk Boneto Instant 700gr
- x_6 : kuantitas produk Boneto Vanilla 700gr
- x_7 : kuantitas produk Boneto Vanilla 350gr
- x_8 : kuantitas produk Indomilk Biokids 4-6 tahun Coklat 800gr
- x_9 : kuantitas produk Indomilk Biokids 4-6 tahun Plain 800gr
- x_{10} : kuantitas produk Indomilk Coklat 800gr
- x_{11} : kuantitas produk Indomilk Plain 800gr
- x_{12} : kuantitas produk S-26 Procal 400gr
- x_{13} : kuantitas produk S-26 Procal 800gr
- x_{14} : kuantitas produk S-26 Promile 400gr
- x_{15} : kuantitas produk S-26 Promile 800gr

4.2.2 Formulasi Fungsi Tujuan

Tujuan yang ingin dicapai dalam tugas akhir ini adalah untuk memaksimalkan profit yang diperoleh minimarket Bentar melalui penataan rak *display* minimarket pada kategori

susu bubuk anak. Berikut merupakan model matematis dari fungsi tujuan berdasarkan layer per layer.

- **Layer 1**

$$\begin{aligned} \text{Maksimum : } Z = & 3425x_1 + 1630x_2 + 615x_3 + 1275x_4 \\ & + 1360x_5 + 2210x_6 + 1985x_7 + 1650x_8 \\ & + 1650x_9 + 1540x_{10} + 1540x_{11} \\ & + 1540x_{12} + 510x_{13} + 845x_{14} + 965x_{15} \\ & + 850x_{16} + 850x_{17} \end{aligned}$$

- **Layer 2**

$$\begin{aligned} \text{Maksimum : } Z = & 1510x_1 + 2600x_2 + 840x_3 + 1675x_4 \\ & + 2327x_5 + 2505x_6 + 4200x_7 + 3890x_8 \\ & + 3285x_9 + 3045x_{10} + 1625x_{11} \\ & + 1775x_{12} + 2275x_{13} + 2275x_{14} \\ & + 4400x_{15} \end{aligned}$$

- **Layer 3**

$$\begin{aligned} \text{Maksimum : } Z = & 840x_1 + 1675x_2 + 4541,25x_3 \\ & + 5675x_4 + 1800x_5 + 1800x_6 \\ & + 2301,25x_7 + 2301,25x_8 + 1800x_9 \\ & + 1800x_{10} + 1800x_{11} + 2301,25x_{12} \\ & + 1800x_{13} + 1800x_{14} + 2240x_{15} \\ & + 2020x_{16} \end{aligned}$$

- **Layer 4**

$$\begin{aligned} \text{Maksimum : } Z = & 945x_1 + 2560x_2 + 1435x_3 + 3295x_4 \\ & + 1745x_5 + 475x_6 + 1485x_7 + 1485x_8 \\ & + 1485x_9 + 1430x_{10} + 1765x_{11} \\ & + 1595x_{12} + 1595x_{13} + 4350x_{14} \\ & + 5950x_{15} + 4200x_{16} \end{aligned}$$

- **Layer 5**

$$\begin{aligned} \text{Maksimum : } Z = & 2485x_1 + 2290x_2 + 3295x_3 + 17455x_4 \\ & + 3295x_5 + 3295x_6 + 1745x_7 + 2925x_8 \\ & + 2925x_9 + 3290x_{10} + 3290x_{11} \\ & + 3600x_{12} + 5850x_{13} + 3000x_{14} \\ & + 5300x_{15} \end{aligned}$$

4.2.3 Formulasi Fungsi Batasan

1. Fungsi Batasan Kuantitas Produk setiap Layer

Batasan kuantitas produk bertujuan untuk membatasi nilai maksimal dan minimal kuantitas produk yang di-*display*. Produk yang di-*display* memiliki batas atas dan batas bawah yang berbeda-beda.

• Layer 1

$$\begin{array}{lll}
 x_1 \geq 0 & x_7 \geq 0 & x_{13} \geq 0 \\
 x_2 \geq 0 & x_8 \geq 0 & x_{14} \geq 0 \\
 x_3 \geq 1 & x_9 \geq 0 & x_{15} \geq 0 \\
 x_4 \geq 1 & x_{10} \geq 0 & x_{16} \geq 0 \\
 x_5 \geq 0 & x_{11} \geq 0 & x_{17} \geq 0 \\
 x_6 \geq 0 & x_{12} \geq 0 &
 \end{array}$$

$$\begin{array}{lll}
 x_1 \leq 1 & x_7 \leq 2 & x_{13} \leq 1 \\
 x_2 \leq 1 & x_8 \leq 1 & x_{14} \leq 1 \\
 x_3 \leq 2 & x_9 \leq 1 & x_{15} \leq 1 \\
 x_4 \leq 2 & x_{10} \leq 1 & x_{16} \leq 1 \\
 x_5 \leq 1 & x_{11} \leq 1 & x_{17} \leq 1 \\
 x_6 \leq 2 & x_{12} \leq 1 &
 \end{array}$$

• Layer 2

$$\begin{array}{lll}
 x_1 \geq 1 & x_6 \geq 0 & x_{11} \geq 0 \\
 x_2 \geq 1 & x_7 \geq 0 & x_{12} \geq 0 \\
 x_3 \geq 0 & x_8 \geq 0 & x_{13} \geq 0 \\
 x_4 \geq 0 & x_9 \geq 0 & x_{14} \geq 0 \\
 x_5 \geq 1 & x_{10} \geq 0 & x_{15} \geq 0
 \end{array}$$

$$\begin{array}{lll}
 x_1 \leq 2 & x_6 \leq 1 & x_{11} \leq 1 \\
 x_2 \leq 2 & x_7 \leq 2 & x_{12} \leq 1 \\
 x_3 \leq 1 & x_8 \leq 1 & x_{13} \leq 1 \\
 x_4 \leq 1 & x_9 \leq 1 & x_{14} \leq 1 \\
 x_5 \leq 2 & x_{10} \leq 1 & x_{15} \leq 1
 \end{array}$$

• Layer 3

$$\begin{array}{lll}
 x_1 \geq 0 & x_4 \geq 1 & x_7 \geq 1 \\
 x_2 \geq 0 & x_5 \geq 0 & x_8 \geq 1 \\
 x_3 \geq 1 & x_6 \geq 0 & x_9 \geq 0
 \end{array}$$

$$\begin{aligned}x_{10} &\geq 0 \\x_{11} &\geq 0 \\x_{12} &\geq 0\end{aligned}$$

$$\begin{aligned}x_{13} &\geq 0 \\x_{14} &\geq 0 \\x_{15} &\geq 0\end{aligned}$$

$$x_{16} \geq 0$$

$$\begin{aligned}x_1 &\leq 1 \\x_2 &\leq 1 \\x_3 &\leq 2 \\x_4 &\leq 2 \\x_5 &\leq 1 \\x_6 &\leq 1\end{aligned}$$

$$\begin{aligned}x_7 &\leq 2 \\x_8 &\leq 2 \\x_9 &\leq 1 \\x_{10} &\leq 1 \\x_{11} &\leq 1 \\x_{12} &\leq 1\end{aligned}$$

$$\begin{aligned}x_{13} &\leq 1 \\x_{14} &\leq 1 \\x_{15} &\leq 1 \\x_{16} &\leq 1\end{aligned}$$

• **Layer 4**

$$\begin{aligned}x_1 &\geq 0 \\x_2 &\geq 0 \\x_3 &\geq 0 \\x_4 &\geq 0 \\x_5 &\geq 0 \\x_6 &\geq 0\end{aligned}$$

$$\begin{aligned}x_7 &\geq 0 \\x_8 &\geq 0 \\x_9 &\geq 0 \\x_{10} &\geq 0 \\x_{11} &\geq 0 \\x_{12} &\geq 0\end{aligned}$$

$$\begin{aligned}x_{13} &\geq 0 \\x_{14} &\geq 0 \\x_{15} &\geq 0 \\x_{16} &\geq 0\end{aligned}$$

$$\begin{aligned}x_1 &\leq 1 \\x_2 &\leq 1 \\x_3 &\leq 1 \\x_4 &\leq 1 \\x_5 &\leq 1 \\x_6 &\leq 1\end{aligned}$$

$$\begin{aligned}x_7 &\leq 1 \\x_8 &\leq 1 \\x_9 &\leq 1 \\x_{10} &\leq 2 \\x_{11} &\leq 2 \\x_{12} &\leq 1\end{aligned}$$

$$\begin{aligned}x_{13} &\leq 1 \\x_{14} &\leq 1 \\x_{15} &\leq 2 \\x_{16} &\leq 1\end{aligned}$$

• **Layer 5**

$$\begin{aligned}x_1 &\geq 0 \\x_2 &\geq 0 \\x_3 &\geq 0 \\x_4 &\geq 0 \\x_5 &\geq 0\end{aligned}$$

$$\begin{aligned}x_6 &\geq 0 \\x_7 &\geq 0 \\x_8 &\geq 0 \\x_9 &\geq 0 \\x_{10} &\geq 0\end{aligned}$$

$$\begin{aligned}x_{11} &\geq 0 \\x_{12} &\geq 0 \\x_{13} &\geq 0 \\x_{14} &\geq 0 \\x_{15} &\geq 0\end{aligned}$$

$$\begin{aligned}x_1 &\leq 1 \\x_2 &\leq 1 \\x_3 &\leq 1 \\x_4 &\leq 1 \\x_5 &\leq 1 \\x_6 &\leq 1\end{aligned}$$

$$\begin{aligned}x_7 &\leq 2 \\x_8 &\leq 1 \\x_9 &\leq 1 \\x_{10} &\leq 2 \\x_{11} &\leq 2 \\x_{12} &\leq 1\end{aligned}$$

$$\begin{aligned}x_{13} &\leq 1 \\x_{14} &\leq 1 \\x_{15} &\leq 1\end{aligned}$$

2. Fungsi Batasan Kapasitas Layer

Batasan kuantitas produk bertujuan untuk membatasi keseluruhan panjang produk yang didisplai agar sesuai dengan panjang rak. Rak yang digunakan pada pengerjaan tugas akhir ini adalah 3 buah rak gondola berukuran 90 cm, sehingga panjang keseluruhan rak yang digunakan adalah 270 cm. Setiap item produk memiliki panjang yang berbeda-beda.

- **Layer 1**

$$15,5x_1 + 15,5x_2 + 9x_3 + 13,5x_4 + 13,5x_5 + 14x_6 + 14x_7 + 14x_8 + 14x_9 + 14x_{10} + 14x_{11} + 14x_{12} + 10,5x_{13} + 12,5x_{14} + 12,5x_{15} + 12x_{16} + 12x_{17} \leq 270$$

- **Layer 2**

$$15,5x_1 + 15,5x_2 + 11x_3 + 15,5x_4 + 16x_5 + 16x_6 + 16x_7 + 14x_8 + 14x_9 + 14x_{10} + 14x_{11} + 14x_{12} + 14x_{13} + 14x_{14} + 20x_{15} \leq 270$$

- **Layer 3**

$$11x_1 + 15,5x_2 + 14x_3 + 14x_4 + 14x_5 + 14x_6 + 14x_7 + 14x_8 + 14x_9 + 14x_{10} + 14x_{11} + 14x_{12} + 14x_{13} + 14x_{14} + 14x_{15} + 14x_{16} \leq 270$$

- **Layer 4**

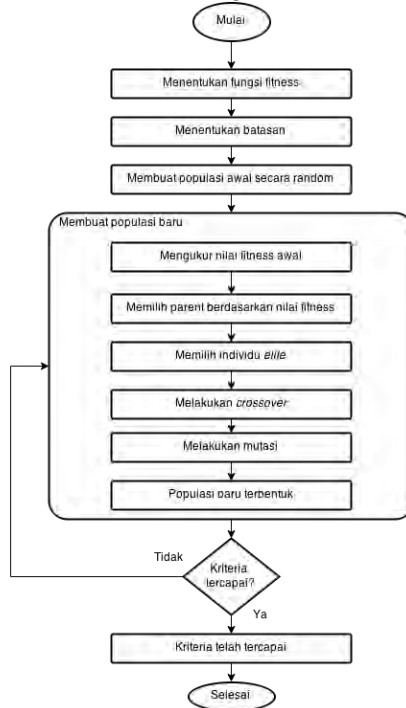
$$14x_1 + 15x_2 + 14x_3 + 16x_4 + 13x_5 + 12x_6 + 14x_7 + 14x_8 + 14x_9 + 14x_{10} + 14x_{11} + 15,5x_{12} + 15,5x_{13} + 16x_{14} + 16x_{15} + 16x_{16} \leq 270$$

- **Layer 5**

$$15x_1 + 15x_2 + 16x_3 + 13x_4 + 16x_5 + 16x_6 + 13x_7 + 16x_8 + 16x_9 + 16x_{10} + 16x_{11} + 16x_{12} + 18x_{13} + 16x_{14} + 18x_{15} \leq 270$$

4.3 Rancangan Algoritma Genetika

Tahapan ini merupakan tahapan proses algoritma genetika untuk mengolah data dengan menggunakan *toolbox* optimasi *ga* pada MATLAB. Pada tahapan ini data proses akan diolah untuk diperoleh hasilnya. Alur kerja dari *toolbox* optimasi *ga* pada Matlab adalah sebagai berikut.



Gambar 4. 1 Proses Algoritma Genetika pada *toolbox ga*

Toolbox optimasi *ga* pada Matlab memiliki beberapa proses yang dilalui sesuai dengan Gambar 4.1. Berikut merupakan penjelasan dalam setiap proses yang dilakukan.

- Menentukan fungsi fitness

Tahap ini menentukan fungsi *fitness* yang bertujuan untuk mengevaluasi sebuah kromosom. Fungsi *fitness* yang baik adalah ketika nilai *fitness* yang tinggi mendominasi suatu populasi. Tujuan dari tugas akhir ini

adalah untuk mendapatkan keuntungan yang optimal dari banyaknya barang yang didisplai pada rak. Maka fungsi *fitness* yang digunakan pada tugas akhir ini adalah sama dengan persamaan (3.1).

- Menentukan batasan

Tahap ini menentukan fungsi batasan yang bertujuan untuk membatasi hasil algoritma genetika sesuai dengan permasalahan yang dialami di Minimarket Bentar sehingga dapat memenuhi kriteria yang telah ditetapkan. Pada tugas akhir ini terdapat dua batasan yang digunakan yaitu batasan panjang rak (Persamaan 4.1) dan batasan kuantitas produk yang didisplai (Persamaan 4.2).

$$\sum_{i=1}^n l_i \times x_{ik} \leq T_k, \text{ dimana } k = 1, \dots, m. \quad (4.1)$$

$$L_i \leq \sum_{k=1}^m x_{ik} \leq U_i, \text{ dimana } i = 1, \dots, n. \quad (4.2)$$

- Membuat populasi awal secara acak

Setelah mendapatkan fungsi *fitness* dan fungsi batasan, algoritma bekerja dimulai dengan membentuk populasi awal secara acak. Populasi dibentuk sesuai dengan jumlah populasi yang ditentukan. Populasi yang dibentuk nantinya terdiri dari kumpulan kromosom. Dalam sebuah kromosom terdapat gen yang mewakili kuantitas setiap produk yang perlu didisplai. Pada tugas akhir ini, jumlah populasi yang digunakan adalah 30, 50, dan 100. Hal ini dilakukan untuk mendapatkan nilai yang paling optimal.

- Membuat populasi baru

Tahap ini merupakan tahap dimana algoritma membentuk serangkaian populasi baru. Pada setiap langkahnya, algoritma menggunakan individu-individu generasi awal untuk membuat populasi baru. Untuk membuat populasi baru, algoritma genetika melakukan beberapa langkah berikut.

- Mengukur nilai *fitness* awal

Tahap ini merupakan tahap dimana algoritma menilai setiap member dari populasi saat ini dengan mengkomputasi nilai *fitness*-nya serta mengukur

nilai *fitness* awal dan mengubahnya menjadi nilai yang lebih berguna.

- Memilih *parent* berdasarkan nilai *fitness*
Tahapan ini merupakan tahapan dimana algoritma melakukan pemilihan *parent* berdasarkan nilai *fitness*-nya. Algoritma memilih sekumpulan dari individu-individu pada sebuah populasi, yaitu *parent*, dimana gen mereka berkontribusi untuk menghasilkan anak-anak. *Parent* yang dipilih biasanya memiliki nilai *fitness* yang lebih baik daripada individu-individu lainnya.
- Memilih individu *elite*
Individu *elite* merupakan individu terbaik pada generasi sebelumnya yang bertahan dalam populasi pada generasi yang baru dan mengikuti proses seleksi pembentukan populasi yang baru.
- Melakukan *crossover*
Tahap ini merupakan tahap dimana *parent* melakukan kawin silang atau pindah silang untuk mendapatkan keturunan. Algoritma akan mempertahankan solusi terbaik dan menghilangkan solusi yang buruk. Skema pindah silang adalah dengan menggabungkan sepasang *parent* dalam populasi saat ini. Fungsi *crossover* standar secara acak memilih entri atau gen pada koordinat yang sama untuk ditukarkan sehingga menjadi anak. Misal peluang pindah silang adalah $P_c = 0,8$ maka diharapkan 80% dari total kromosom akan mengalami pindah silang. Fungsi pindah silang yang dilakukan pada pengerjaan tugas akhir ini adalah dengan menggunakan fungsi *single point crossover* yang terdapat pada *toolbox* optimasi *ga* Matlab. Fungsi ini memilih satu titik pada kromosom *parent* secara acak lalu menggabungkannya dengan kromosom *parent* lainnya.

- Melakukan mutasi

Proses mutasi merupakan proses dimana dilakukan perubahan secara acak pada *parent* tunggal. Algoritma membentuk anak mutasi secara acak dengan mengubah gen dari *parent* masing-masing. Mutasi diperlukan untuk mengembalikan informasi titik poin dari kromosom yang hilang akibat proses *crossover*. Probabilitas mutasi (P_m) lebih baik memiliki nilai yang kecil agar tidak menghasilkan individu yang lemah akibat seringnya dilakukan konfigurasi pada gen unggul. Menurut Haupt (2000), probabilitas mutasi yang baik yaitu diantara 5% hingga 20% [18]. Pada pengerjaan tugas akhir ini, mutasi dilakukan dengan menggunakan fungsi adaptasi kelayakan yang merupakan fungsi mutasi standar untuk permasalahan yang memiliki batasan dan fungsi *uniform*. Fungsi kelayakan adaptif menghasilkan arahan yang adaptif secara acak terhadap generasi yang sukses maupun tidak sukses sedangkan pada fungsi *uniform*, algoritma memilih probabilitas mutasi untuk menentukan banyaknya individu yang dilakukan mutasi lalu mengganti setiap individu yang terpilih dengan angka acak yang dipilih secara seragam.

- Populasi baru terbentuk

Setelah melalui proses *crossover* dan mutasi maka didapatkanlah individu-individu baru yang merupakan hasil dari perpindahan silang dan mutasi dari *parent* yang telah dipilih. Individu-individu ini akan membentuk populasi baru bersama dengan individu elite.

- Kriteria telah tercapai

Proses pembuatan populasi baru akan terus berulang hingga algoritma mendapatkan hasil yang sesuai dengan kriteria yang telah ditentukan. Selain itu algoritma dapat berhenti sesuai dengan kondisi yang ditentukan. Kondisi

ini salah satunya adalah banyaknya generasi. Algoritma akan berhenti ketika jumlah generasi mencapai nilai yang ditentukan, misalkan 100 [9].

BAB V IMPLEMENTASI

Bab ini menjelaskan mengenai implementasi bentuk matematis algoritma genetika ke dalam *toolbox* optimasi *ga* pada Matlab serta pengolahan hasil optimal algoritma genetika dengan menggunakan pemrograman integer.

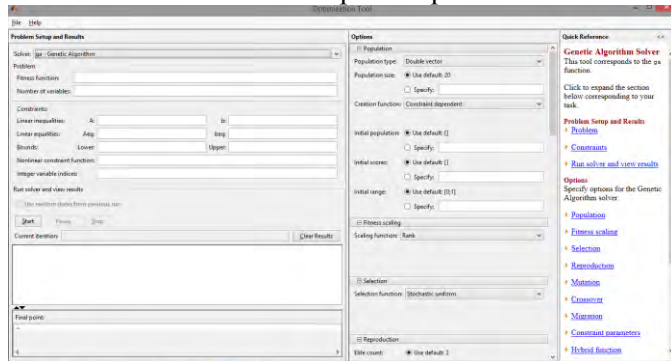
5.1 Implementasi Algoritma Genetika

Algoritma genetika diimplementasikan dengan menggunakan *toolbox* optimasi yang terdapat pada program Matlab. Fungsi matematis yang telah dibuat dituliskan di sebuah file *.m* pada Matlab. *Toolbox* optimasi dapat diakses melalui *command window* Matlab dengan mengetikkan '*optimtool('ga')*' lalu menekan tombol Enter.



Gambar 5. 1 Memanggil *toolbox* optimasi *ga*

Kemudian akan muncul tampilan seperti Gambar 5.2.



Gambar 5. 2 Tampilan *toolbox* optimasi *ga*

Pada *toolbox* optimasi *ga* terdapat 2 kolom yang diperlukan untuk mengimplementasikan algoritma genetika, yaitu kolom *Problem Setup and Results* di bagian kiri dan

kolom *Options* di bagian kanan. Kolom *Problem Setup and Results* berisi fungsi tujuan dan batasan dari permasalahan serta hasil dari algoritma genetika. Sedangkan kolom *Options* berisi konfigurasi fungsi-fungsi yang digunakan pada algoritma genetika, seperti mutasi, pindah silang, jumlah populasi dan lainnya. Berikut merupakan penjelasan bagian-bagian dari *toolbox* optimasi *ga* pada Matlab.

- Kolom *Problem Setup and Results*

1) *Problem*

Gambar 5. 3 Problem

Pada bagian ini terdapat dua baris yang wajib diisi untuk melakukan optimasi dengan menggunakan algoritma genetika yaitu *fitness function* dan *number of variables*. Berikut merupakan penjelasan dari *fitness function* dan *number of variables*.

- *Fitness function*, merupakan fungsi tujuan dari permasalahan yang ingin dioptimasi. Baris ini berisi nama file *fitness function* yang telah dibuat sebelumnya. Dalam pengerjaan tugas akhir ini file *fitness function* dibuat menjadi lima layer yaitu layer 1 (Formula 5.1), layer 2 (Formula 5.2), layer 3 (Formula 5.3), layer 4 (Formula 5.4) dan layer 5 (Formulasi 5.5).

```
function y = fitness_layer1(x)
y = -1*(3425*x(1)+1630*x(2)+615*x(3)+1275*x(4)+1360*x(5)+2210*x(6)+ ...
1985*x(7)+1650*x(8)+ 1650*x(9)+1540*x(10)+1540*x(11)+1540*x(12)+...
510*x(13)+845*x(14)+965*x(15)+850*x(16)+850*x(17)) ;
end
```

Formula 5. 1 Fitness function rak - Layer 1

```
function y = fitness_layer2(x)
y = -1*(1510*x(1)+2600*x(2)+840*x(3)+1675*x(4)+2327*x(5)+2505*x(6)+4200*x(7)+...
3890*x(8)+3285*x(9)+3045*x(10)+1625*x(11)+1775*x(12)+2275*x(13)+2275*x(14)+4400*x(15)) ;
end
```

Formula 5. 2 Fitness function rak - Layer 2

```
function y = fitness_layer3(x)
y = -1*(840*x(1)+1675*x(2)+4541.25*x(3)+5675*x(4)+1800*x(5)+1800*x(6)+2301.25*x(7)+ ...
2301.25*x(8)+1800*x(9)+1800*x(10)+1800*x(11)+2301.25*x(12)+1800*x(13)+1800*x(14)+...
2240*x(15)+2020*x(16)) ;
end
```

Formula 5.3 Fitness function rak - Layer 3

```
function y = fitness_layer4(x)
y = -1*(945*x(1)+2560*x(2)+1435*x(3)+3295*x(4)+1745*x(5)+475*x(6)+1485*x(7)+...
1485*x(8)+ 1485*x(9)+1430*x(10)+1765*x(11)+1595*x(12)+1595*x(13)+4350*x(14)+...
5950*x(15)+4200*x(16)) ;
end
```

Formula 5.4 Fitness function rak - Layer 4

```
function y = fitness_layer5(x)
y = -1*(2485*x(1)+2290*x(2)+3295*x(3)+1745*x(4)+3295*x(5)+3295*x(6)+1745*x(7)+2925*x(8)+...
2925*x(9)+3290*x(10)+3290*x(11)+3600*x(12)+5850*x(13)+3000*x(14)+5300*x(15)) ;
end
```

Formula 5.5 Fitness function rak - Layer 5

- *Number of variables*, baris ini berisi jumlah variabel independen pada fungsi *fitness*.

2) Constraints

Constraint berfungsi untuk memberikan batasan pada hasil optimasi algoritma genetika. Batasan yang diberikan dapat berupa batasan linier maupun batasan nonlinier. Berikut merupakan macam-macam batasan yang diberikan oleh *toolbox* optimasi *ga*.

- *Linear inequalities*, yaitu merupakan batasan dari suatu fungsi yang berbentuk $A_x \leq b$ atau $A_x < b$.
- *Linear equalities*, merupakan batasan dari suatu fungsi yang berbentuk persamaan $Aeq*x = beq$ dimana keduanya ditulis dalam bentuk matrix *Aeq* dan vektor *beq*.
- *Bounds*, merupakan batasan bawah (*Lower*) dan batasan atas (*Upper*) dari sebuah variabel. Misalkan $0 \leq x \leq 2$.
- *Nonlinear constraint function*, merupakan batasan *nonlinear* dari suatu fungsi yang tertulis dalam sebuah file .m. Pada fungsi batasan ini, kesamaan nonlinear merupakan bentuk dari $ceq = 0$ dan ketidaksamaan nonlinear merupakan bentuk dari $c \leq 0$.

- *Integer variable indices*, merupakan batasan yang menunjukkan bahwa komponen x bernilai bulat.

The image shows a software interface for defining constraints. It has several input fields:

- Linear inequalities:** A: [] b: []
- Linear equalities:** Aeq: [] beq: []
- Bounds:** Lower: [] Upper: []
- Nonlinear constraint function:** []
- Integer variable indices:** []

Gambar 5. 4 Constraints

Pada pengerjaan tugas akhir ini terdapat dua batasan yang digunakan yaitu batasan *Bounds* dan batasan *nonlinear constraint function*. Setiap layer memiliki batasan yang berbeda-beda seperti yang telah dijelaskan pada Bab 4 sebelumnya. Batasan *nonlinear constraint function* dari setiap layer dapat dilihat pada Formula 5.6, Formula 5.7, Formula 5.8, Formula 5.9, dan Formula 5.10.

```
function [c, ceq] = constrain_layerr1(x)

c = 15.5*x(1)+15.5*x(2)+9*x(3)+13.5*x(4)+13.5*x(5)+14*x(6)+14*x(7)+14*x(8)+...
    14*x(9)+14*x(10)+14*x(11)+14*x(12)+10.5*x(13)+12.5*x(14)+12.5*x(15)+12*x(16)+...
    12*x(17)-270;
ceq=[];
end
```

Formula 5. 6 Nonlinear constraint function pada Layer 1

```
function [c, ceq] = constrain_layerr2(x)

c = 15.5*x(1)+15.5*x(2)+11*x(3)+15.5*x(4)+16*x(5)+16*x(6)+16*x(7)+...
    14*x(8)+14*x(9)+14*x(10)+14*x(11)+14*x(12)+14*x(13)+14*x(14)+20*x(15)-270;
ceq=[];
end
```

Formula 5. 7 Nonlinear constraint function pada Layer 2

```
function [c, ceq] = constrain_layerr3(x)

c = 11*x(1)+15.5*x(2)+14*x(3)+14*x(4)+14*x(5)+14*x(6)+14*x(7)+14*x(8)+...
    14*x(9)+14*x(10)+14*x(11)+14*x(12)+14*x(13)+14*x(14)+14*x(15)+14*x(16)-270;
ceq=[];
end
```

Formula 5. 8 Nonlinear constraint function pada Layer 3

```
function [c, ceq] = constrain_layerr4(x)

c = 14*x(1)+15*x(2)+14*x(3)+16*x(4)+13*x(5)+12*x(6)+14*x(7)+14*x(8)+...
    14*x(9)+14*x(10)+14*x(11)+15.5*x(12)+15.5*x(13)+16*x(14)+16*x(15)+16*x(16)-270;
ceq=[];
end
```

Formula 5. 9 Nonlinear constraint function pada Layer 4

```
function [c, ceq] = constrain_layer5(x)

c = (15*x(1)+15*x(2)+16*x(3)+13*x(4)+16*x(5)+16*x(6)+13*x(7)+16*x(8)+...
16*x(9)+16*x(10)+16*x(11)+16*x(12)+18*x(13)+16*x(14)+18*x(15))-270;
ceq=[];
end
```

Formula 5. 10 Nonlinear constraint function pada Layer 5

3) Run Solver and view results

Gambar 5.5 merupakan kolom yang menunjukkan hasil optimasi *ga* yang telah dilakukan. Bagian ini menunjukkan dua baris yaitu *current iteration* dan *final point*. Berikut merupakan penjerelasan dari *current iteration* dan *final point*.

- *Current iteration*, menunjukkan jumlah iterasi yang telah dilakukan oleh program untuk menyelesaikan satu permasalahan.
- *Final Point*, menunjukkan hasil maksimum yang keluar sebagai variabel x dari fungsi *fitness* yang dimiliki setelah mengklik 'start'.



Gambar 5. 5 Run Solver and view results

- Kolom *Options*
Kolom *options* berfungsi untuk menspesifikasikan pilihan pada *solver* algoritma genetika sesuai dengan yang diinginkan. Berikut merupakan beberapa bagian pada kolom *options* yang digunakan.
 - 1) *Population*, kolom ini menspesifikasikan pilihan populasi dalam algoritma genetika. Tipe populasi yang digunakan dalam tugas akhir ini adalah *double vector* sedangkan besarnya populasi yang

digunakan adalah 30, 50, dan 100 [9] [18]. Gambar 5.6 menunjukkan kolom populasi.

The image shows a software interface for configuring a population. It has several sections with radio buttons for 'Use default' and text boxes for 'Specify'.

- Population type:** Double vector (dropdown menu)
- Population size:** ☒ Use default: 20; ☐ Specify: [text box]
- Creation function:** Constraint dependent (dropdown menu)
- Initial population:** ☒ Use default: []; ☐ Specify: [text box]
- Initial scores:** ☒ Use default: []; ☐ Specify: [text box]
- Initial range:** ☒ Use default: [0;1]; ☐ Specify: [text box]

Gambar 5. 6 Population

- 2) *Fitness Scaling*, kolom ini berfungsi untuk memilih metode *scaling fitness* yaitu mengubah nilai *fitness* yang mentah menjadi nilai *fitness* yang sesuai dengan fungsi seleksi. Metode *fitness scaling* yang digunakan adalah 'Rank' yaitu menilai nilai *fitness* berdasarkan tingkatan tiap individu. Tingkatan dari seorang individu adalah posisinya dalam urutan nilai. Individu yang bernilai *fitness* paling tinggi akan bernilai 1 dan seterusnya. Gambar 5.7 menunjukkan kolom *fitness scaling* dengan menggunakan metode 'Rank'.

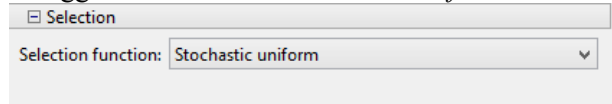
The image shows a software interface for configuring fitness scaling. It has a dropdown menu for the scaling function.

- Scaling function:** Rank (dropdown menu)

Gambar 5. 7 Fitness scaling

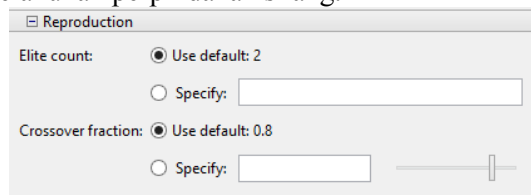
- 3) *Selection*, merupakan kolom yang berfungsi untuk memilih metode yang digunakan dalam proses pemilihan *parent* untuk generasi selanjutnya berdasarkan *fitness scaling* yang dipilih sebelumnya. Metode *selection* yang digunakan

adalah *Stochastic Uniform*. Metode ini menjabarkan sebuah baris dimana setiap *parent* berkorespondensi terhadap bagian dari panjang garis yang sebanding dengan nilai skala. Gambar 5.8 menunjukkan kolom *Selection* dengan menggunakan metode '*Stochastic uniform*'.



Gambar 5. 8 Selection

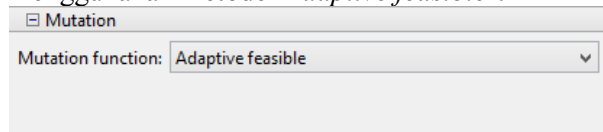
- 4) *Reproduction*, kolom ini menentukan bagaimana algoritma genetika membuat anak di setiap generasi yang baru. Gambar 5.9 menunjukkan kolom *Reproduction* yang terdiri dari dua baris *Elite count* dan *Crossover fraction*. *Elite count* menspesifikasikan jumlah individu yang terjamin untuk hidup di generasi selanjutnya. Jumlah *elite count* harus bilangan bulat yang kurang atau sama dengan ukuran populasi. *Crossover fraction* menspesifikasikan probabilitas pindah silang dari generasi selanjutnya. *Crossover fraction* digunakan untuk menentukan nilai P_c yang diinginkan pada proses pindah silang. Individu atau *parent* yang memiliki nilai dibawah P_c akan terpilih untuk melakukan perpindahan silang.



Gambar 5. 9 Reproduction

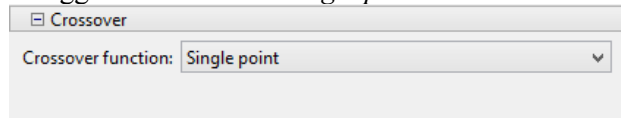
- 5) *Mutation*, merupakan kolom berfungsi untuk memilih metode mutasi. Fungsi mutasi membuat perubahan acak yang kecil pada individu-individu di populasi dimana memberikan keragaman genetika dan memungkinkan algoritma genetika

untuk mencari solusi di ruang yang lebih luas. Metode mutasi yang digunakan adalah *Adaptive feasible* dan *uniform*. Metode *adaptive feasible* merupakan fungsi standar untuk permasalahan yang memiliki batasan. Metode ini menghasilkan arahan secara acak yang adaptif terhadap generasi yang sukses ataupun generasi yang gagal. Metode *uniform* memberikan probabilitas mutasi dimana probabilitas tersebut digunakan untuk menentukan jumlah individu yang akan dimutasi lalu algoritma akan mengganti setiap individu yang terpilih dengan angka acak yang dipilih secara seragam berdasarkan nilai awal suatu individu. Gambar 5.10 menunjukkan kolom *Mutation* dengan menggunakan metode '*Adaptive feasible*'.



Gambar 5. 10 Mutation

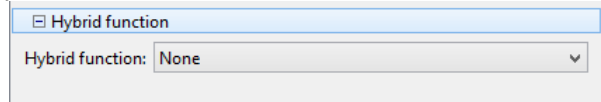
- 6) *Crossover*, kolom ini berfungsi untuk memilih metode pindah silang pada *parent* ataupun dua individu untuk membentuk individu baru pada generasi selanjutnya. Metode pindah silang yang digunakan pada tugas akhir ini adalah metode *Single Point* dimana perpindahan silang terjadi pada satu titik yang dipilih secara acak. Gambar 5.11 menunjukkan kolom *Crossover* dengan menggunakan metode '*Single point*'.



Gambar 5. 11 Crossover

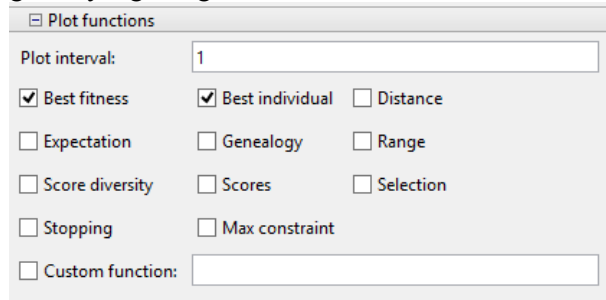
- 7) *Hybrid function*, kolom ini memiliki fungsi untuk memilih satu metode lain untuk dicampur dengan algoritma genetika. Fungsi *hybrid* merupakan

fungsi optimasi lain yang berjalan setelah algoritma genetika berakhir. Metode *hybrid* yang digunakan dalam pengerjaan tugas akhir ini adalah *fmincon* yaitu fungsi optimasi untuk permasalahan dengan batasan. Gambar 5.12 menunjukkan kolom *Hybrid function*.



Gambar 5. 12 Hybrid function

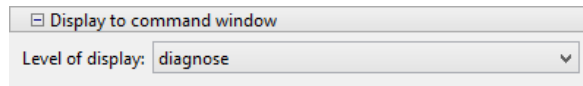
- 8) *Plot functions*, kolom ini mengatur grafik apa saja yang ingin ditampilkan ketika *solver* berjalan. Contohnya grafik *best fitness*, *best individual* dan lainnya. Gambar 5.13 menunjukkan kolom *Plot functions* yang digunakan untuk menampilkan grafik yang diinginkan.



Gambar 5. 13 Plot functions

- 9) *Display to command window*, kolom ini mengatur informasi apa saja yang ingin ditampilkan pada *command window* Matlab ketika *solver* dilakukan. Display yang ditawarkan oleh kolom ini antara lain adalah *Off* – tidak mendisplay apa pun, *iterative* – menampilkan informasi setiap iterasi dari algoritma, *diagnose* – menampilkan informasi setiap iterasi dan informasi permasalahan serta adanya perubahan pada opsi standar, dan *final* – menampilkan alasan mengapa *solver* berhenti.

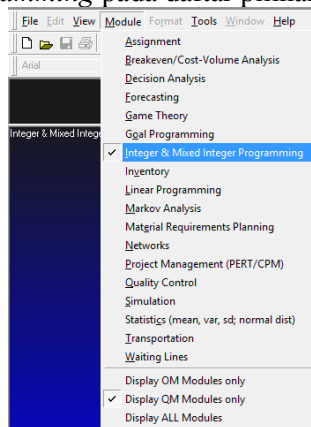
Kolom *Display to command window* ditunjukkan pada Gambar 5.14.




Gambar 5. 14 *Display to command window*

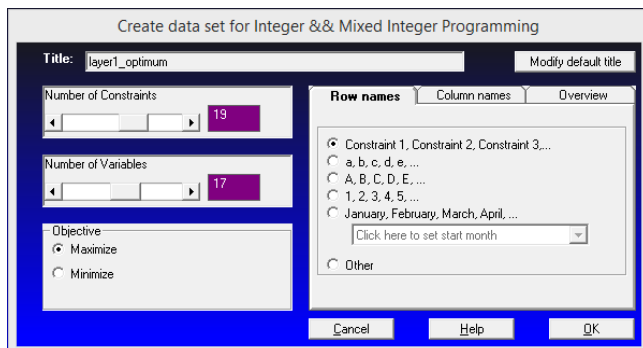
5.2 Implementasi Pemrograman Integer

Pada tugas akhir ini, pemrograman integer digunakan untuk membulatkan hasil riil algoritma genetika sehingga didapatkan hasil yang optimal. Metode ini diimplementasikan dengan menggunakan bantuan modul *integer & mixed integer programming* pada *software POM-QM for windows*. Untuk melakukan implementasi pemrograman integer dengan modul tersebut yang harus dilakukan adalah memilih modul *integer & mixed integer programming* pada daftar pilihan modul.



Gambar 5. 15 Inisialisasi POM-QM

Langkah selanjutnya adalah membuat file *.mix* untuk memulai optimasi menggunakan modul tersebut dengan mengklik ikon () pada *toolbar*. Maka akan muncul tampilan seperti pada Gambar 5.16.



Gambar 5. 16 Membuat data set pada modul integer & mixed integer programming


Optimasi dapat dimulai dengan mengisi kolom-kolom yang telah disediakan. Berikut merupakan penjelasan beberapa kolom yang harus diisi.

- *Title*, merupakan nama file .mix yang berisi permasalahan yang ingin diselesaikan.
- *Number of Constraints*, berisi jumlah batasan yang diperlukan untuk menyelesaikan suatu permasalahan.
- *Number of Variables*, berisi jumlah variabel independen yang digunakan pada permasalahan yang ingin diselesaikan.
- *Objective*, memberikan dua pilihan tujuan optimasi yaitu untuk memaksimalkan (*Maximize*) atau untuk meminimalkan (*Minimize*) suatu permasalahan.

Setelah mengisi kolom-kolom tersebut klik *Ok*, maka akan muncul tampilan seperti Gambar 5.17. Kolom-kolom tersebut lalu diisi dengan fungsi tujuan dan batasan-batasan yang ada pada permasalahan. Berikut merupakan fungsi tujuan dan batasan permasalahan dari hasil optimal algoritma genetika pada layer 1.

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	X17	RHS	Equation form			
Maximize	3420	1850	815	1275	1360	2210	1880	1680	1840	1640	1540	110	840	960	880	860	270	15.340X1 + 15.002X2				
Constraint 1	15.5	15.5	8	13.5	14	14	14	14	14	14	14	14	14	12.5	12.5	12	12	270	15.5X1 + 15.5X2 ≤ 270			
Constraint 2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	X1 ≤ 1			
Constraint 3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	X2 ≤ 1			
Constraint 4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	X3 ≤ 1			
Constraint 5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	X4 ≤ 1			
Constraint 6	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	X5 ≤ 1			
Constraint 7	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	X6 ≤ 1			
Constraint 8	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	X7 ≤ 1			
Constraint 9	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	X8 ≤ 1			
Constraint 10	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	X9 ≤ 1			
Constraint 11	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	X10 ≤ 1			
Constraint 12	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	X11 ≤ 1			
Constraint 13	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	X12 ≤ 1			
Constraint 14	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	X13 ≤ 1			
Constraint 15	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	X14 ≤ 1			
Constraint 16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	X15 ≤ 1			
Constraint 17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	X16 ≤ 1			
Constraint 18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	X17 ≤ 1		
Constraint 19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	X18 ≤ 1	
Variable Type	Integer	Integer	Integer	Integer	Integer	Integer	Integer	Integer	Integer	Integer	Integer	Integer	Integer	Integer	Integer	Integer	Integer					

Gambar 5. 17 Data set permasalahan pada rak - layer 1

Klik ikon ( Solve) untuk memulai melakukan optimasi. Optimasi akan melakukan iterasi beberapa kali untuk mendapatkan hasil paling optimal. Gambar 5.18 merupakan *output* dari hasil pengerjaan pemrograman integer.

Variable	Type	Value
X1	Integer	1
X2	Integer	1
X3	Integer	2
X4	Integer	2
X5	Integer	1
X6	Integer	2
X7	Integer	2
X8	Integer	1
X9	Integer	1
X10	Integer	1
X11	Integer	1
X12	Integer	1
X13	Integer	0
X14	Integer	1
X15	Integer	1
X16	Integer	1
X17	Integer	1
Solution value		30015

Gambar 5. 18 Output dari integer programming

Maka hasil bulat dari algoritma genetika telah didapatkan setelah menggabungkannya dengan pemrograman integer. Hasil tersebut nantinya akan menjadi acuan untuk banyaknya alokasi produk pada rak *display* Minimarket Bentar cabang Delta Sari Sidoarjo.

BAB VII

KESIMPULAN DAN SARAN

Bab ini menguraikan tentang kesimpulan yang dapat diambil selama proses pengerjaan tugas akhir yang telah dilakukan. Dalam bab ini diuraikan pula saran untuk pengembangan lebih lanjut dari tugas akhir ini.

7.1 Kesimpulan

Adapun beberapa hal yang dapat disimpulkan dalam pengerjaan Tugas Akhir ini sebagai berikut :

1. Pengoptimasian alokasi produk perlu dilakukan untuk memperoleh keuntungan yang optimal.
2. Pemilihan jumlah populasi dan nilai probabilitas pindah silang memiliki peran penting dalam optimasi algoritma genetika.
3. Jumlah populasi (N_{pop}) yang dapat memberikan hasil paling optimal pada permasalahan alokasi produk pada rak minimarket adalah 100 untuk semua layer.
4. Nilai probabilitas pindah silang (P_c) yang dapat memberikan hasil paling optimal pada permasalahan alokasi produk pada rak minimarket adalah 0,3 untuk layer 1, 0,4 untuk layer 2, 0,1 untuk layer 3, 0,2 untuk layer 4, dan 0,4 untuk layer 5.
5. Fungsi mutasi yang sesuai dengan permasalahan alokasi produk dan memberikan hasil paling optimal adalah fungsi mutasi *adaptive feasible*.
6. Penggunaan fungsi *hybrid* dapat memberikan hasil yang paling optimal dan konsisten pada optimasi algoritma genetika.
7. Algoritma genetika memberikan total keuntungan yang lebih tinggi dibandingkan algoritma genetika yang dimodifikasi dengan pemrograman integer

namun kurang sesuai dengan permasalahan yang dihadapi oleh Minimarket Bentar.

8. Total keuntungan optimal untuk setiap layer adalah sebagai berikut.
 - Layer 1 sejumlah Rp 30.015,00 dengan menggunakan fungsi *hybrid fmincon* untuk seluruh nilai N_{pop} dan P_c dimana fungsi mutasi yang digunakan adalah *adaptive feasible*.
 - Layer 2 sejumlah Rp 46.514,00 dengan menggunakan fungsi *hybrid fmincon* untuk seluruh nilai N_{pop} dan P_c dimana fungsi mutasi yang digunakan adalah *adaptive feasible*.
 - Layer 3 sejumlah Rp 50.473,75 dengan menggunakan fungsi *hybrid fmincon* untuk seluruh nilai N_{pop} dan P_c dimana fungsi mutasi yang digunakan adalah *adaptive feasible*.
 - Layer 4 sejumlah Rp 44.465,00 dengan menggunakan fungsi *hybrid fmincon* untuk seluruh nilai N_{pop} dan P_c dimana fungsi mutasi yang digunakan adalah *adaptive feasible*.
 - Layer 5 sejumlah Rp 54.910,00 dengan menggunakan fungsi *hybrid fmincon* untuk seluruh nilai N_{pop} dan P_c dimana fungsi mutasi yang digunakan adalah *adaptive feasible*.

7.2 Saran

Adapun saran yang dapat digunakan untuk mengembangkan Tugas Akhir ini selanjutnya. Pada penelitian selanjutnya, diharapkan pengoptimasian alokasi produk dapat mempertimbangkan jumlah produk yang tersedia di gudang, permintaan pelanggan dan frekuensi pembelian pelanggan. Sehingga kinerja rak *display* dapat lebih optimal dan dapat menyesuaikan kebutuhan pelanggan.

DAFTAR PUSTAKA

- [1] Darandono, "SWA Online," SWA, 26 Februari 2012. [Online]. Available: <http://swa.co.id/business-strategy/markerting/2012-bisnis-ritel-di-indonesia-kian-menjanjikan>. [Diakses 20 Oktober 2014].
- [2] ahm, "765 Ribu Gerai Ritel Menjamur di Indonesia - Liputan6.com," 30 Januari 2014. [Online]. Available: <http://bisnis.liputan6.com/read/814452/765-ribu-gerai-ritel-menjamur-di-indonesia>.
- [3] M. L. Pandin, "Potret Bisnis Ritel di Indonesia: Pasar Modern," *Economic Review*, pp. 90-95, 2009.
- [4] M.-C. Chen dan C.-P. Lin, "A data mining approach to product assortment and shelf space allocation," *Expert Systems with Application*, pp. 976-986, 2007.
- [5] H. Hwang, B. Choi dan G. Lee, "A genetic algorithm approach to an integrated problem of shelf space design and item allocation," *Computer & Industrial Engineering*, pp. 809-820, 2009.
- [6] P. Larson dan R. DeMarais, "Psychic stock: An independent variable category of inventory," *International Journal of Physical Distribution and Logistics Management*, vol. 20, no. 7, pp. 28-34, 1990.
- [7] M. Nafari dan J. Shahrabi, "A temporal data mining approach for shelf-space allocation with consideration product price," *Expert Systems with Applications*, pp. 4066-4072, 2010.
- [8] S. Raut, S. Swami dan M. Moholkar, "Heuristic and meta-heuristica approaches for multi-period shelf-space optimization: the case of motion picture retailing," *Journal of the Operational Research Society*, vol. 60, no. 10, pp. 1335-1348, 2009.

- [9] M. Castelli dan L. Vanneschi, "Genetic algorithm with variable neighborhood search for the optimal allocation for goods in shop shelves," *Operation Research Letters*, vol. 42, pp. 355-360, 2014.
- [10] M. Levy dan B. A. Weitz, *Retailing Management Seventh Edition*, New York: McGraw-Hill/Irwin, 2009.
- [11] B. Berman dan J. R. Evans, *Retail Management: A Strategic Approach*, New Jersey: Prentice Hall, 1998.
- [12] A. Sujana, *Paradigma Baru dalam Modern Retail Management*, Jakarta: WPA ReSULTANT, 2009.
- [13] C. W. Utami, *Manajemen Ritel : Strategi dan Implementasi Operasional Bisnis Ritel Modern di Indonesia*, Jakarta: Salemba Empat, 2010.
- [14] E. Turban, R. E. Sharda dan D. Delen, *Decision Support and Business Intelligence Systems*, 9th Edition, New Jersey: Prentice Hall, 2010.
- [15] R. L. Haupt dan S. E. Haupt, *Practical Genetic Algorithms Second Edition*, New Jersey: John Wiley & Sons, Inc., 2004.
- [16] B. W. Taylor, *Introduction to Management Science Ninth Edition*, New Jersey: Pearson Prentice Hall, 2007.
- [17] E. N. Hayati, "Aplikasi Algoritma Branch and Bound Untuk Menyelesaikan Integer Programming," *Dinamika Teknik*, vol. IV, no. 1, pp. 13-23, 2010.
- [18] R. L. Haupt, "Optimum Population Size and Mutation Rate for Simple Real Genetic Algorithm that Optimizes Array Factors," *Antennas and Propagation Society International Symposium, IEEE.*, vol. 2, no. 2, pp. 1034 - 1037, 2000.
- [19] R. P. Indreswari, *Optimasi Dynamic Pricing Menggunakan Algoritma Genetika Menggunakan Metode Algoritma Genetika Berdasarkan Model Permintaan Pada Hotel JW Marriott Surabaya*, Surabaya: ITS, 2013.

LAMPIRAN A

KETERANGAN DATA PRODUK

No	Nama produk	Berat	Ukuran (cm)	
			panjang	tinggi
1	Bebelove 1	400 gr	15,5	20
2	Bebelove 2	400 gr	15,5	20
3	Bebelac 3	400 gr	15,5	20
4	Bebelac 4	400 gr	15,5	20
5	Bimbi 1	150 gr	14	17
6	Bimbi 1	400 gr	15	19
7	Bimbi 2	400 gr	15	19
8	Bimbi 3	400 gr	15	19
9	Bimbi Lola	130 gr	14	17
10	Boneto Coklat	700 gr	16	21
11	Boneto Coklat	350 gr	13	18
12	Boneto Coklat	150 gr	12	15
13	Boneto Honey	700 gr	16	21
14	Boneto Honey	350 gr	13	18
15	Boneto Instant	700 gr	16	21
16	Boneto Vanila	700 gr	16	21
17	Boneto Vanila	350 gr	13	18
18	Indomilk Biokids 1-3 tahun	400 gr	14	20,5
19	Indomilk Biokids 4-6 tahun Coklat	400 gr	14	20,5
20	Indomilk Biokids 4-6 tahun Coklat	800 gr	16	24
21	Indomilk Biokids 4-6 tahun Plain	400 gr	14	20,5

No	Nama produk	Berat	Ukuran (cm)	
			panjang	tinggi
22	Indomilk Biokids 4-6 tahun Plain	800 gr	16	24
23	Indomilk Coklat	400 gr	14	20,5
24	Indomilk Coklat	800 gr	16	24
25	Indomilk Plain	400 gr	14	20,5
26	Indomilk Plain	800 gr	16	24
27	Lactona 1	200 gr	11	17
28	Lactona 1	400 gr	15,5	21
29	Lactona 2	200 gr	11	17
30	Lactona 2	400 gr	15,5	21
31	Lactona 3 Madu	400 gr	15,5	21
32	Lactona 3 Vanilla	400 gr	15,5	21
33	Milo	150 gr	9	16
34	Milo	300 gr	13,5	20,5
35	Milo	600gr	16	25
36	Milo 3 in 1	300 gr	13,5	20,5
37	Milo 3 in 1	600 gr	16	25
38	Morinaga BMT	200 gr	14	19
39	Morinaga BMT	400 gr	14	21
40	Morinaga Chil Mil	200 gr	14	19
41	Morinaga Chil Mil	400 gr	14	21
42	Morinaga Chil School Platinum	800 gr	14	21
43	Morinaga Chil School Platinum	800 gr	14	21
44	Morinaga Chil-Kid Madu	200 gr	14	19
45	Morinaga Chil-Kid Madu	400 gr	14	21
46	Morinaga Chil-Kid Vanilla	200 gr	14	19

No	Nama produk	Berat	Ukuran (cm)	
			panjang	tinggi
47	Morinaga Chil-School Coklat	200 gr	14	19
48	Morinaga Chil-School Regular	400 gr	14	21
49	Morinaga Chil-School Stroberi	200 gr	14	19
50	Morinaga Chil-School Vanilla	200 gr	14	19
51	Ovaltine	150 gr	10,5	18
52	Ovaltine	280 gr	12,5	20,5
53	Ovaltine	600 gr	14	23
54	Ovaltine 3 in 1	300 gr	12,5	20,5
55	Ovaltine 3 in 1	600 gr	14	23
56	S-26 (1)	400 gr	16	19
57	S-26 Nutrisure Gold	400 gr	16	19
58	S-26 Procal (3)	400 gr	16	19
59	S-26 Procal (3)	700 gr	18	20
60	S-26 Promil (2)	400 gr	16	19
61	S-26 Promise (4)	400 gr	16	19
62	S-26 Promise (4)	700 gr	18	20
63	Vitalac Madu	150 gr	12	18
64	Vitalac Madu	400 gr	14	20
65	Vitalac Vanilla	150 gr	12	18
66	Vitalac Vanilla	400 gr	14	20
67	Vitalac Vanilla	800 gr	20	25
68	Zee Regular Swizz Chocolate Twist	350 gr	14	21
69	Zee Regular Strawberry Milk	350 gr	14	21
70	Zee Platinum Vanilla Delight	350 gr	14	21

No	Nama produk	Berat	Ukuran (cm)	
			panjang	tinggi
71	Zee Platinum Choco Rich	350 gr	14	21
72	Zee Kidz 5+ Vanilla	350 gr	14	21
73	Zee Kidz 5+ Honey	350 gr	14	21
74	Zee Kidz 5+ Chocolate Milk	350 gr	14	21
75	Zee Kidz Platinum Vanilla Delight	350 gr	14	21
76	Zee Tween Chocolate Milk	350 gr	14	21
77	Zee Tween Strawberry	350 gr	14	21
78	Zee Tween Platinum Choco	350 gr	14	21
79	Zee Tween Platinum Straw ice	350 gr	14	21

ABCDEFGHIJ
KLMNOPQRST
UVWXYZ

TEST TEST TEST TEST TEST TEST TEST TEST TEST
HALO COBA HALO COBA

LAMPIRAN B OPTIMASI ALGORITMA GENETIKA

LAYER 1

1. Hasil Rata-rata Uji Coba

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	Total Profit
Pop = 30	Pm = AF	Pc = 0,1	1	0,9979	1,9308	1,9915	0,9979	1,9989	1,9997	0,9994	0,9989	0,9994	0,9983	0,9975	0,828	0,9227	0,9811	0,9408	0,9668	30201,9
		Pc = 0,2	1	0,9975	1,9311	1,9898	0,9962	1,9999	1,9996	0,9992	0,9991	0,9988	0,998	0,9982	0,8507	0,9479	0,9732	0,9528	0,9228	30197,1
		Pc = 0,3	1	0,998	1,923	1,9819	0,9938	1,9999	1,9999	0,9992	0,9984	0,9949	0,9979	0,9977	0,8604	0,9285	0,9721	0,9619	0,9482	30188,9
		Pc = 0,4	1	0,9958	1,9321	1,987	0,9901	2	1,9983	0,9986	0,9991	0,9959	0,997	0,9974	0,8605	0,9419	0,9732	0,9395	0,9524	30186,3
		Pc = 0,5	1	0,9978	1,9296	1,80859	0,9936	2	1,9993	0,9989	0,9951	0,9978	0,994	0,9974	0,8319	0,9304	0,9739	0,9702	0,9554	29964,5
		Pc = 0,6	1	0,9934	1,936	1,9779	0,9858	1,9995	1,9972	0,9938	0,9933	0,9898	0,9941	0,9967	0,8716	0,9495	0,9841	0,9481	0,9545	30163,3
		Pc = 0,7	1	0,9956	1,8866	1,9868	0,963	1,9956	1,9976	0,997	0,9891	0,9887	0,995	0,9937	0,9033	0,9488	0,9792	0,9706	0,9683	30144,1
		Pc = 0,8	1	0,9893	1,9416	1,9612	0,9836	1,996	1,992	0,9926	0,9939	0,9859	0,9893	0,9875	0,8934	0,9557	0,9885	0,9688	0,965	30136,3
		Pc = 0,9	1	0,9903	1,9604	1,9805	0,9885	1,9962	1,9966	0,9877	0,9887	0,9697	0,9922	0,9889	0,8565	0,9739	0,9726	0,9771	0,9584	30138

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	Total Profit
Pop = 50	Pm = AF	Pc = 0,1	1	0,9992	1,9232	1,9908	0,9984	1,9979	2	0,9989	1	0,9988	0,9986	0,9986	0,8165	0,9539	0,9745	0,9827	0,9226	30212,3
		Pc = 0,2	1	0,9989	1,9206	1,8938	0,9974	2	1,9993	0,9995	0,999	0,9983	0,9986	0,9994	0,8262	0,9342	0,9801	0,942	0,9557	30075,5
		Pc = 0,3	1	0,9977	1,9482	1,9915	0,9983	1,9998	1,9997	0,9985	0,9996	0,9992	0,999	0,9721	0,799	0,9569	0,9806	0,9526	0,9576	30192,4
		Pc = 0,4	1	0,9981	1,9266	1,9975	0,9983	2	2	0,998	0,9988	0,9984	0,9976	0,9975	0,8348	0,9433	0,9661	0,9256	0,9704	30202,7
		Pc = 0,5	1	0,9987	1,9214	1,9953	0,9972	2	1,9985	0,998	0,9984	0,9982	0,9982	0,9995	0,8557	0,9415	0,9571	0,9449	0,9535	30198,7
		Pc = 0,6	1	0,9981	1,8991	1,9853	0,9863	1,999	2	0,9963	0,9957	0,9987	0,9966	0,9974	0,8679	0,9363	0,9851	0,954	0,9345	30165,5
		Pc = 0,7	1	0,9919	1,9226	1,992	0,9902	2	1,9996	0,9978	0,9939	0,9978	0,9904	0,9919	0,868	0,9438	0,9655	0,9499	0,9659	30175,9
		Pc = 0,8	1	0,9945	1,9163	1,9779	0,9918	1,9976	2	0,9973	0,9956	0,9936	0,9985	0,9959	0,8909	0,9467	0,9507	0,9587	0,9564	30169,3
		Pc = 0,9	1	0,986	1,9484	1,81009	1,0639	1,9972	1,9869	0,9923	0,9944	0,9846	0,9817	0,9879	0,9372	0,9307	0,9745	0,9621	0,9727	30019,9

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	Total Profit
Pop = 100	Pm = AF	Pc = 0,1	1	0,9992	1,928	1,9992	0,999	2	2	0,9994	1	0,9995	0,9988	0,9997	0,7442	0,9399	0,9951	0,9724	0,9626	30231,4
		Pc = 0,2	1	0,9995	1,8892	1,9991	0,9987	2	1,9993	0,9997	0,999	0,9991	1	1	0,7523	0,9747	0,9888	0,9579	0,9689	30228,4
		Pc = 0,3	1	0,9995	1,944	1,9981	0,9989	1,9999	1,9995	0,9993	1	0,9993	0,9992	0,9994	0,7234	0,9409	0,9938	0,9666	0,9765	30234,9
		Pc = 0,4	1	0,9988	1,9123	1,9963	0,9982	2	1,9994	0,9988	1	0,9984	0,9988	0,9985	0,7272	0,9679	0,9837	0,9741	0,9811	30232,1
		Pc = 0,5	1	0,9989	1,9045	1,9977	0,9983	2	2	0,9991	0,9994	0,9993	0,999	0,9987	0,7597	0,9441	0,9868	0,9842	0,9639	30225,5
		Pc = 0,6	1	0,9991	1,9175	1,9937	0,9982	1,9992	2	0,9984	0,9991	0,9984	0,9994	0,9988	0,7723	0,9555	0,9796	0,9572	0,9728	30218,3
		Pc = 0,7	1	0,9975	1,9102	1,9923	0,9968	1,9995	1,9996	0,9992	1	0,9965	0,9969	0,9972	0,7934	0,9502	0,9737	0,9542	0,9769	30202,4
		Pc = 0,8	1	0,9968	1,9329	1,8882	0,9935	2	1,9987	0,9966	0,9976	0,9984	0,9985	0,9971	0,8342	0,9461	0,9805	0,9289	0,9663	30067,7
		Pc = 0,9	1	0,9934	1,9524	1,9915	0,9837	2	1,9963	0,9904	0,9926	0,984	0,9882	0,9818	0,8467	0,9671	0,9804	0,9649	0,9606	30157,5

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	Total Profit
Pop = 30	Pm = 0,1	Pc = 0,1	0,991	0,9822	1,9624	1,9708	0,9789	1,0807	1,0856	0,9778	0,9819	0,9829	0,9722	0,962	0,9647	0,9861	0,9764	0,9741	0,9693	26235,61
	Uniform	Pc = 0,2	0,985	0,9821	1,9767	1,9819	0,9787	0,9821	0,9837	0,9812	0,9723	0,9707	0,9861	0,9703	0,9702	0,9758	0,9728	0,9819	0,978	25827,56
		Pc = 0,3	0,993	0,9878	1,9867	1,9868	0,9816	0,9775	0,9811	0,9659	0,9662	0,9773	0,9787	0,9667	0,9775	0,9782	0,9828	0,9701	0,9808	25827,85
		Pc = 0,4	0,992	0,9792	1,9746	1,9776	0,9883	0,9817	0,9887	0,9902	0,9822	0,9867	0,978	0,9865	0,9833	0,9795	0,9817	0,9858	0,9722	25944,13
		Pc = 0,5	0,99	0,988	1,9865	1,984	0,987	0,9857	0,9775	0,9863	0,9844	0,99	0,9813	0,9875	0,9568	0,9717	0,9756	0,9918	0,9879	25953,63
		Pc = 0,6	0,989	0,9915	1,979	1,9657	0,9687	0,987	0,9745	0,9869	0,9829	0,985	0,986	0,9834	0,9847	0,9771	0,9745	0,9773	0,9849	25895,96
		Pc = 0,7	0,984	0,97518	1,9701	1,981	0,9784	0,9762	0,9833	0,9646	0,9816	0,9757	0,9742	0,9668	0,976	0,9851	0,9769	0,9774	0,9788	25775,94
		Pc = 0,8	0,982	0,972	1,9729	1,9818	0,9741	0,9779	0,9717	0,9795	0,9514	0,9824	0,9805	0,9748	0,9595	0,9632	0,9816	0,9713	0,9852	25722,75
		Pc = 0,9	0,98	0,9547	1,9654	1,9594	0,9602	0,9636	0,8499	0,9574	0,9715	0,9734	0,9748	0,9475	0,9661	0,8459	0,9728	0,9658	0,9729	25460,96

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	Total Profit
Pop = 50	Pm = 0,1	Pc = 0,1	0,982	0,987	1,9762	1,981	0,9741	0,9812	0,9822	0,9756	0,9798	0,9824	0,977	0,9843	0,9834	0,9741	0,9795	0,9844	0,986	25879,09
	Uniform	Pc = 0,2	0,992	0,9872	1,9851	1,886	0,9779	0,9925	0,988	0,9894	0,9888	0,9824	0,9844	0,9882	0,9724	0,9845	0,9757	0,9777	0,991	25874,62
		Pc = 0,3	0,995	0,988	1,9844	1,9832	0,986	0,987	0,9879	0,9789	0,9827	0,9843	0,9896	0,9939	0,9835	0,9859	0,9918	0,9828	0,9917	26027,99
		Pc = 0,4	0,994	0,9932	1,9836	1,8865	0,9852	0,993	0,9848	0,9801	0,9863	0,9838	0,9931	0,9907	0,99	0,9834	0,9866	0,984	0,9842	25912
		Pc = 0,5	0,993	0,9861	1,994	1,9934	0,9928	0,9936	0,9893	0,9889	0,9864	0,9833	0,9759	0,9876	0,9894	0,9901	0,9874	0,9926	0,9907	26061,57
		Pc = 0,6	0,989	0,9865	1,9841	1,9858	0,9936	0,9838	0,9903	0,9837	0,9826	0,9844	0,9859	0,9724	0,9877	0,9875	0,9901	0,9888	0,9886	25990,26
		Pc = 0,7	0,993	0,9893	1,9894	1,9844	0,983	0,9874	0,9874	0,9913	0,9857	0,9874	0,9913	0,9857	0,9859	0,9821	0,9891	0,986	0,9944	25978,7
		Pc = 0,8	0,984	0,9926	1,9842	1,9816	0,9821	0,9869	0,9857	0,9835	0,965	0,9811	0,9655	0,9869	0,9791	0,9878	0,9854	0,9795	0,9905	25898,97
		Pc = 0,9	0,984	0,9851	1,9832	1,9708	0,9758	0,9806	0,9643	0,9747	0,9813	0,9795	0,9842	0,9812	0,9724	0,9793	0,9756	0,9777	0,9851	25813,2

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	Total Profit
Pop = 100	Pm = 0,1	Pc = 0,1	0,99	0,9886	1,9889	1,9901	0,9884	0,9888	0,9913	0,9935	0,984	0,9824	0,9868	0,9884	0,9884	0,9882	0,9875	0,9878	0,9856	26048,31
	Uniform	Pc = 0,2	0,996	0,9942	1,9926	1,9962	0,9905	0,9818	0,9892	0,9901	0,9938	0,9919	0,9918	0,993	0,9844	0,9875	0,9891	0,9913	0,9904	26114,12
		Pc = 0,3	0,998	0,9949	1,9835	1,9917	0,9915	0,9917	0,9956	0,9954	0,9913	0,9953	0,9913	0,9929	0,989	0,9836	0,9904	0,9954	0,9919	26169,85
		Pc = 0,4	0,993	0,9971	1,992	1,994	0,9941	0,9957	0,9952	0,9921	0,9928	0,9944	0,9941	0,9932	0,9954	0,995	0,9929	0,994	0,9935	26184,19
		Pc = 0,5	0,991	0,9946	1,9929	1,9956	0,9937	0,9919	0,9937	0,9932	0,9892	0,9907	0,9891	0,9916	0,992	0,9901	0,9911	0,9948	0,9943	26139,41
		Pc = 0,6	0,996	0,997	1,9917	1,9925	0,9882	0,993	0,9929	0,9896	0,992	0,9933	0,9953	0,9911	0,9935	0,992	0,9933	0,9963	0,9935	26163,27
		Pc = 0,7	0,995	0,9951	1,9944	1,9964	0,991	0,9954	0,9953	0,9953	0,9898	0,9919	0,9949	0,9933	0,9953	0,9907	0,9945	0,9956	0,9966	26189,06
		Pc = 0,8	0,993	0,9904	1,9929	1,9919	0,9958	0,9958	0,989	0,9887	0,9929	0,9914	0,9959	0,9925	0,9937	0,9878	0,9854	0,9927	0,9927	26138,39
		Pc = 0,9	0,988	0,9924	1,9897	1,9878	0,9703	0,9765	0,9896	0,987	0,9828	0,9883	0,9832	0,9932	0,9828	0,9772	0,9889	0,9882	0,9922	25983,42

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	Total Profit	
Pop = 30	Pm = 0,05	Pc = 0,1	0,989	0,9865	1,9712	1,9846	0,9792	0,9811	0,9779	0,9802	0,9788	0,9854	0,9728	0,9867	0,9887	0,9839	0,9873	0,9816	0,9805	25902,71
		Uniform	Pc = 0,2	0,988	0,9892	1,9823	1,983	0,983	0,9915	0,9849	0,9853	0,9806	0,9838	0,9853	0,9852	0,9751	0,9753	0,9851	0,982	0,983
		Pc = 0,3	0,988	0,9824	1,9861	1,985	0,9854	0,9883	0,9846	0,9784	0,9803	0,9815	0,9875	0,9863	0,9677	0,9781	0,985	0,9826	0,9857	25941,51
		Pc = 0,4	0,987	0,9833	1,9735	1,9897	0,9833	0,9864	0,9885	0,9852	0,9721	0,983	0,9899	0,9902	0,983	0,9803	0,9869	0,9858	0,9815	25958,82
		Pc = 0,5	0,985	0,985	1,98	1,9891	0,9833	0,9887	0,9809	0,9856	0,9681	0,983	0,9905	0,9838	0,9837	0,9857	0,9829	0,9866	0,9698	25924,98
		Pc = 0,6	0,989	0,9845	1,9747	1,9741	0,9806	0,9852	1,0159	0,9927	0,9866	0,9908	0,9747	0,9863	0,9833	0,9771	0,9814	0,9862	0,9867	26012,34
		Pc = 0,7	0,986	0,9845	1,9572	1,975	0,9741	0,985	0,9777	0,9761	0,9785	0,9761	0,9584	0,9865	0,9506	0,9697	0,9805	0,9713	0,9828	25780,06
		Pc = 0,8	0,987	0,982	1,9767	1,9767	0,9757	0,978	0,9783	0,9793	0,9772	0,9489	0,9801	0,9814	0,9676	0,963	0,9796	0,9734	0,9727	25762,94
		Pc = 0,9	0,98	0,9689	1,954	1,9633	0,9648	0,9629	0,9604	0,9628	0,9815	0,967	0,9567	0,9861	0,9587	0,9792	0,9727	0,9295	0,9737	25393,81

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	Total Profit
Pop = 50	Pm = 0,05	Pc = 0,1	0,994	0,9908	1,9866	1,9861	0,8997	0,9923	0,985	0,9851	0,9845	0,9908	0,9834	0,9824	0,9848	0,9871	0,991	0,9802	0,9927	26038,61
		Uniform	Pc = 0,2	0,993	0,9846	1,9816	1,9861	0,991	0,9901	0,9922	0,9841	0,9913	0,9915	0,9916	0,9832	0,9858	0,9889	0,9869	0,9896	0,9911
		Pc = 0,3	0,988	0,991	1,9844	1,9847	0,9926	0,9907	0,9931	0,9918	0,9859	0,9886	0,9913	0,9858	0,9883	0,9864	0,9941	0,9905	0,9933	26072,91
		Pc = 0,4	0,996	0,992	1,9902	1,992	0,9927	0,9932	0,9956	0,9908	0,9882	0,9901	0,9835	0,9824	0,9857	0,993	0,9941	0,9932	0,9884	26115,57
		Pc = 0,5	0,988	0,9929	1,9874	1,9876	0,9912	1,0576	0,9943	0,9917	0,9868	0,991	0,9895	0,9907	0,9888	0,9883	0,995	0,9904	0,9928	26244,62
		Pc = 0,6	0,99	0,9908	1,9903	1,9921	0,9946	0,991	0,992	0,9866	0,9795	0,9826	0,9868	0,9912	0,9912	0,9891	0,9926	0,9856	0,9896	26062,73
		Pc = 0,7	0,988	0,9947	1,9855	1,9901	0,9925	0,988	0,9942	0,9872	0,9863	0,9929	0,9875	0,9928	0,9758	0,9906	0,9842	0,9916	0,9948	26078,74
		Pc = 0,8	0,989	0,9814	1,9924	1,987	0,9814	0,9849	0,9771	0,9786	0,9785	0,9784	0,9788	0,9847	0,9716	0,9883	0,9838	0,9799	0,985	25907,91
		Pc = 0,9	0,995	0,9812	1,9826	1,9924	0,9689	0,9786	0,9756	0,9814	0,9787	0,9506	0,9853	0,9846	0,9685	0,98	0,9676	0,9634	0,9925	25834,18

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

Uji coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	Total Profit
Pop = 100	Pm = 0,05	Pc = 0,1	0,996	0,995	1,9959	1,9933	0,9948	0,9944	0,9959	0,9942	0,9947	0,9927	0,9958	0,996	0,9936	0,9919	0,9951	0,9965	0,9958	26207,3
		Pc = 0,2	0,994	0,9962	1,9917	1,9953	0,9947	0,9946	0,9885	0,9936	0,9921	0,9948	0,9968	0,9933	0,9931	0,9933	0,9946	0,9921	0,9955	26175,7
		Pc = 0,3	0,993	0,9944	1,9927	1,9957	0,9951	0,9959	0,9925	0,9937	0,9947	0,995	0,9959	0,9956	0,9933	0,9871	0,9977	0,9915	0,9957	26188,28
		Pc = 0,4	0,996	0,9964	1,9983	1,9964	0,9955	0,997	0,9966	0,9967	0,9969	0,9972	0,9965	0,9947	0,9964	0,9942	0,9932	0,9912	0,9945	26229,71
		Pc = 0,5	0,997	0,9961	1,997	1,994	0,998	0,9961	0,9944	0,9953	0,9908	0,9937	0,995	0,9963	0,993	0,9911	0,9929	0,9967	0,996	26210,55
	Uniform	Pc = 0,6	0,994	0,987	1,9928	1,9924	0,9944	0,9969	0,9933	0,993	0,9955	0,9939	0,9966	0,9959	0,993	0,9956	0,992	0,9905	0,9948	26179,98
		Pc = 0,7	0,996	0,9962	1,9902	1,9932	0,9974	0,9886	0,9907	0,9915	0,9954	0,9953	0,9923	0,9892	0,9933	0,995	0,9971	0,9954	0,9883	26166,24
		Pc = 0,8	0,992	0,9923	1,9927	1,9932	0,9917	0,9892	0,99	0,9929	0,9933	0,9907	0,9892	0,9914	0,9886	0,9927	0,9924	0,9957	0,9907	26122,33
		Pc = 0,9	0,988	0,9902	1,9745	1,9825	0,9892	0,982	0,9894	0,9859	0,9823	0,9906	0,9838	0,9926	0,9915	0,9898	0,9891	0,99173	0,9849	26013,48

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	Total Profit
Pop = 30	Pm = AF	Pc = 0,1	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,2	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,3	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,4	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,5	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
	Hybrid	Pc = 0,6	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,7	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,8	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,9	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	Total Profit
Pop = 50	Pm = AF	Pc = 0,1	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,2	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,3	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,4	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,5	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
	Hybrid	Pc = 0,6	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,7	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,8	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,9	1	1	2	2	1	2	2	1	1	1	1	1	0,524	1	1	1	1	30282,2

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	Total Profit
Pop = 100	Pm = AF	Pc = 0,1	1	1	2	2	1	2	2	1	1	1	1	0,524	1	1	1	1	30282,2
	Hybrid	Pc = 0,2	1	1	2	2	1	2	2	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,3	1	1	2	2	1	2	2	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,4	1	1	2	2	1	2	2	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,5	1	1	2	2	1	2	2	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,6	1	1	2	2	1	2	2	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,7	1	1	2	2	1	2	2	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,8	1	1	2	2	1	2	2	1	1	1	1	0,524	1	1	1	1	30282,2
		Pc = 0,9	1	1	2	2	1	2	2	1	1	1	1	0,524	1	1	1	1	30282,2

LAYER 2

1. Hasil Rata-rata Uji Coba

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit	
1	Pop = 30	Pm = AF	Pc = 0,1	1,7552	1,9932	0,586	0,8474	1,9504	0,9861	2	0,9995	0,9996	0,9991	0,8027	0,8798	0,9869	0,9983	1	47149,4
2			Pc = 0,2	1,839	1,999	0,5218	0,8177	1,9595	0,9942	2	0,9999	1	0,9997	0,7534	0,9003	0,9836	0,9954	0,9996	47173,9
3			Pc = 0,3	1,6967	1,9962	0,4888	0,8255	1,97078	0,986	2	1	0,9997	0,9995	0,8807	0,9688	0,9779	0,9804	0,9998	47225,9
4			Pc = 0,4	1,761	1,998	0,4899	0,855	1,9712	0,955	2	1	0,9968	0,9973	0,8703	0,8898	0,9912	0,9904	1	47179,8
5			Pc = 0,5	1,8258	1,9957	0,48467	0,8064	1,9774	0,9727	1,9991	0,9994	1	1	0,899	0,8851	0,9488	0,9642	0,9999	47139,1
6			Pc = 0,6	1,7689	1,8881	0,7794	0,7504	1,8475	0,914	1,9999	1	0,9938	0,9935	0,8317	0,935	0,9756	0,9741	0,9982	46498,6
7			Pc = 0,7	1,8568	1,9637	0,7222	0,8804	1,919	0,9511	1,9977	0,9985	0,9893	0,9859	0,7363	0,8851	0,9651	0,9797	1	46957,1
8			Pc = 0,8	1,8496	1,9867	0,5737	0,8392	1,9569	0,9643	1,9969	0,9756	0,9794	0,9809	0,8926	0,8344	0,9706	0,9778	1	46965,5
9			Pc = 0,9	1,9	1,994	0,825	0,836	1,955	0,892	1,972	0,915	0,966	0,988	0,915	0,934	0,894	0,917	1	46618,7

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit	
1	Pop = 50	Pm = AF	Pc = 0,1	1,7473	1,9972	0,3483	0,8354	1,9813	0,9995	2	1	0,9997	0,9983	0,8826	0,9604	0,9862	0,9938	0,9993	47291,4
2			Pc = 0,2	1,7393	1,9994	0,3465	0,8295	1,9835	1	2	1	0,9994	0,9998	0,9249	0,9269	0,9828	0,9987	0,9998	47298,5
3			Pc = 0,3	1,695	1,9971	0,4332	0,8655	1,9807	0,9887	2	1	0,9997	0,9984	0,8407	0,958	0,9986	0,9963	0,9998	47269,5
4			Pc = 0,4	1,6147	1,9996	0,5347	0,8382	1,9845	0,9983	2	1	0,9993	1	0,884	0,9608	0,9818	0,9961	1	47268,3
5			Pc = 0,5	1,7402	2	0,4845	0,8337	1,9893	0,9967	2	0,9994	1	0,9995	0,8291	0,9154	0,9924	0,9893	0,9994	47251
6			Pc = 0,6	1,7883	1,9947	0,5649	0,7753	1,9734	0,998	1,9993	0,9996	1	0,9994	0,7949	0,9262	0,9754	0,9903	1	47173,2
7			Pc = 0,7	1,841	1,9971	0,46	0,8351	1,9663	0,9416	2	0,9996	1	0,9986	0,8831	0,8828	0,9616	0,9956	1	47160,7
8			Pc = 0,8	1,7614	1,9929	0,6169	0,8047	1,9327	0,9725	2	0,9988	0,9972	0,999	0,8579	0,8718	0,9869	0,9573	1	47008,5
9			Pc = 0,9	1,8478	1,9566	0,7604	0,8805	1,9069	0,9421	1,9806	0,9834	0,9673	0,9708	0,8594	0,8752	0,977	0,9327	0,9907	46719,5

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
1	Pop = 100	Pm = AF	Pc = 0,1	1,7587	1,9997	0,2289	0,851	1,9992	0,9994	1,9998	1	1	0,9994	0,9199	0,9412	0,9995	0,9991	1	47357,8
2			Pc = 0,2	1,668	1,9991	0,309	0,8342	1,997	0,9987	2	1	1	0,9996	0,9404	0,9843	0,999	0,9994	1	47362,4
3			Pc = 0,3	1,7778	1,9993	0,2094	0,8685	1,9979	0,9989	2	1	0,999	1	0,9234	0,9181	0,9992	0,9997	1	47359
4			Pc = 0,4	1,7615	1,9989	0,10024	0,9365	1,9981	0,9987	2	1	0,9995	0,9997	0,8897	0,9744	0,9995	0,999	1	47400,5
5			Pc = 0,5	1,7591	1,9996	0,3513	0,8538	1,9993	0,9985	2	1	0,9995	0,9999	0,836	0,9392	0,9992	0,9986	0,9999	47322,2
6			Pc = 0,6	1,7645	1,9984	0,3145	0,8675	1,9948	0,9983	2	1	0,9993	0,9999	0,8589	0,924	0,9981	0,999	1	47316,7
7			Pc = 0,7	1,7737	1,999	0,355	0,866	1,9906	0,995	2	0,9996	1	0,9995	0,8401	0,9216	0,9849	0,9988	0,9998	47279
8			Pc = 0,8	1,7895	1,9967	0,3244	0,8639	1,9833	0,9844	2	1	0,9996	0,9997	0,873	0,9131	0,9971	0,9952	0,9997	47282,5
9			Pc = 0,9	1,7532	1,9767	0,6957	0,7981	1,73984	0,9521	1,9997	1	0,9985	0,9958	0,8533	0,9268	0,9771	0,9821	1	46631,5

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
1	Pop = 30	Pm = 0,1	Pc = 0,1	1,9668	1,9747	0,9707	0,9899	1,8774	0,9835	0,9832	0,9825	0,992	0,9826	0,9851	0,9844	0,9797	0,9887	0,9862	43777,5
2		Uniform	Pc = 0,2	1,9859	1,9879	0,980	0,969	1,985	0,983	0,9757	0,9869	0,9842	0,9767	0,989	0,980	0,983	0,980	0,9912	44012,2
3			Pc = 0,3	1,9674	1,9773	0,9003	0,9728	1,9877	0,9836	1,1489	0,984	0,9777	0,971	0,9458	0,9728	0,9852	0,9892	0,9787	44470
4			Pc = 0,4	1,9882	1,9912	0,9841	0,9832	1,9799	0,9878	0,9906	0,9858	0,9854	0,9798	0,9892	0,9826	0,9843	0,9881	0,9898	44144,2
5			Pc = 0,5	1,989	1,987	0,9901	0,987	1,989	0,984	1,087	0,987	0,9932	0,9821	0,986	0,9905	0,990	0,985	0,9906	44618,8
6			Pc = 0,6	1,984	1,9888	0,9707	0,9798	1,9778	0,9775	0,9841	0,9901	0,9843	0,9905	0,9865	0,98	0,9884	0,9868	0,9888	44095,3
7			Pc = 0,7	1,9863	1,9874	0,9757	0,9858	1,8866	1,0841	0,988	0,9839	0,9894	0,9821	0,9827	0,9856	0,9789	0,9902	0,9824	44109,4
8			Pc = 0,8	1,983	1,9847	0,9808	0,9772	1,9821	0,9701	0,9874	0,9809	0,9834	0,9803	0,9852	0,9823	0,9869	0,9812	0,9873	44002
9			Pc = 0,9	1,9614	1,9805	0,9646	0,9717	1,78976	0,9666	0,9622	0,976	0,9736	0,9696	0,9621	0,9679	0,9687	0,9559	0,9753	43074,8

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
1	Pop = 50	Pm = 0,1	Pc = 0,1	1,9816	1,9902	0,9832	0,9831	1,8809	1,0834	0,98	0,9911	0,9867	0,9898	0,9843	0,9756	0,9871	0,9884	0,9909	44142,1
2		Uniform	Pc = 0,2	1,9805	1,9853	0,9839	0,9902	1,9906	0,9861	1,023	0,9839	0,9901	0,9865	0,9843	0,9883	0,9936	0,9777	0,9939	44331,6
3			Pc = 0,3	1,9951	1,9916	0,9812	0,9908	1,9886	0,9887	0,9896	0,9873	0,9897	0,9867	0,9862	0,99	0,9936	0,9933	0,9946	44287,6
4			Pc = 0,4	1,9901	1,9916	0,9929	0,9861	1,9892	0,9886	0,9907	0,9918	0,989	0,9873	0,9926	0,9919	0,9902	0,9894	0,9929	44294,5
5			Pc = 0,5	1,9938	1,9888	0,9867	0,9845	1,9922	0,9881	0,98649	0,9886	0,9861	0,9876	0,9839	0,9947	0,9874	0,991	0,9935	44242,6
6			Pc = 0,6	1,9953	1,99	0,9843	0,9928	1,9847	0,99	0,9956	0,9879	0,988	0,9908	0,9881	0,9935	0,9875	0,9919	0,9929	44303
7			Pc = 0,7	1,9924	1,987	0,9799	0,9872	1,9922	0,9851	0,9918	0,9912	0,9826	0,9811	0,9898	0,9895	0,9862	0,9855	0,9915	44204,5
8			Pc = 0,8	1,9789	1,9879	0,9338	0,9572	1,9829	0,9884	1,0774	0,9821	0,9882	0,9896	0,9892	0,9763	0,9886	0,991	0,9926	44451
9			Pc = 0,9	1,9879	1,9843	0,9776	0,9755	1,9789	0,9731	0,9727	0,9817	0,9858	0,9684	0,9803	0,9825	0,9732	0,975	0,9714	43793,1

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
1	Pop = 100	Pm = 0,1	Pc = 0,1	1,9903	1,9939	0,9916	0,9918	1,989	0,9886	0,9904	0,9901	0,9954	0,9934	0,9945	0,9956	0,9916	0,9913	0,9947	44365,5
2		Uniform	Pc = 0,2	1,9914	1,9907	0,9945	0,9943	1,9909	0,994	0,9958	0,9962	0,9941	0,9921	0,9941	0,9951	0,9911	0,9943	0,9944	44424,5
3			Pc = 0,3	1,9943	1,9951	0,9938	0,9944	1,9879	0,9912	0,992	0,9964	0,9909	0,9941	0,9889	0,9937	0,9937	0,9942	0,9963	44409,4
4			Pc = 0,4	1,9966	1,9926	0,996	1,0928	1,8963	0,9946	0,9965	0,9955	0,9946	0,9946	0,9958	0,9924	0,9934	0,9955	0,9973	44413
5			Pc = 0,5	1,9945	1,996	0,9934	0,9941	1,992	0,9932	0,9975	0,9959	0,9955	0,9969	0,9964	0,9932	0,9941	0,9939	0,997	44485,1
6			Pc = 0,6	1,996	1,994	0,9931	1,0944	1,71615	0,9944	0,9951	0,9931	0,992	0,9948	0,9913	0,9922	0,9957	0,9958	0,9952	43962,2
7			Pc = 0,7	1,9953	1,9969	0,9945	0,9924	1,9901	0,9967	0,9957	0,9903	0,9926	0,995	0,994	0,9965	0,994	0,9932	0,9975	44448,8
8			Pc = 0,8	1,975	1,998	0,999	0,988	1,992	0,993	0,993	0,996	0,993	0,997	0,996	0,99	0,991	0,988	0,988	44362
9			Pc = 0,9	1,994	1,992	0,9895	0,9903	1,9849	0,9872	0,9924	0,9924	0,9908	0,98482	0,98427	0,9839	0,9922	0,9904	0,9948	44287,3

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
1	Pop = 30	Pm = 0,05	Pc = 0,1	1,98736	1,99479	0,97268	0,98885	1,98131	0,98258	0,98344	0,9927	0,99356	0,98012	0,991	0,98353	0,9876	0,98991	0,99439	44203,3
2			Pc = 0,2	1,9825	1,9856	0,988	0,989	1,992	0,982	0,9903	0,9957	0,987	0,9864	0,987	0,986	0,987	0,974	0,9906	44189,5
3			Pc = 0,3	1,9685	1,8927	0,9872	0,9886	1,9888	0,9917	0,9884	0,9949	0,9161	0,9876	0,9923	0,9909	0,9949	0,9892	0,9851	43748,4
4			Pc = 0,4	1,9882	1,9737	0,9896	0,9733	1,81581	0,9915	0,9565	0,986	0,995	0,9869	0,9834	0,9884	0,9887	0,9882	0,9919	43645,2
5			Pc = 0,5	1,991	1,981	0,980	0,983	1,987	0,983	0,981	0,986	0,9905	0,981	0,991	0,9831	0,977	0,991	0,9927	44113,7
6			Pc = 0,6	1,9834	1,99	0,9703	0,9772	1,89	0,9913	0,9936	0,9907	0,9805	0,9852	0,983	0,9853	0,9832	0,9872	0,9784	43883,8
7			Pc = 0,7	1,9773	1,9882	0,9849	0,9865	1,9873	0,9643	0,97	0,9802	0,9819	0,9838	0,9686	0,9822	0,9723	0,9776	0,9877	43882,2
8			Pc = 0,8	1,989	1,9735	0,956	0,9734	1,9857	0,9683	0,9803	0,9749	0,9692	0,9851	0,9866	0,9853	0,9875	0,9843	0,9815	43863,9
9			Pc = 0,9	1,9509	1,9153	0,9705	0,9381	1,95	0,961	0,9634	0,9557	0,9667	0,9648	0,9666	0,9607	0,966	0,9659	0,9587	43023,8

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
1	Pop = 50	Pm = 0,05	Pc = 0,1	1,99135	1,98959	0,98959	0,98854	1,9837	0,99629	0,99095	0,98705	0,9939	0,99284	0,99263	0,98914	0,99355	0,99153	0,99357	44325
2		Uniform	Pc = 0,2	1,9846	1,81446	0,98993	0,98255	1,99301	0,9867	0,98254	0,99086	0,99329	0,99146	0,9942	0,99179	0,99561	0,99394	0,99612	43849,3
3			Pc = 0,3	1,99399	1,98556	0,98609	0,9906	1,9889	0,99016	0,99014	0,99588	0,99012	0,99385	0,99547	0,99209	0,99628	0,99183	0,99285	44350,9
4			Pc = 0,4	1,99314	1,89	0,99042	0,99385	1,98688	0,99301	0,98904	0,9914	0,99584	0,99079	0,99107	0,99201	0,99012	0,98869	0,99499	44081,1
5			Pc = 0,5	1,9904	1,98926	0,9916	0,99096	1,9879	0,98714	0,98533	0,99056	0,99389	0,98773	0,99256	0,99317	0,9888	0,98834	0,98678	44248,8
6			Pc = 0,6	1,9891	1,9927	0,995	0,9894	1,9885	0,9832	0,9904	0,9961	0,9957	0,9914	0,9944	0,991	0,9928	0,9879	0,9928	44341,3
7			Pc = 0,7	1,9831	1,994	0,994	0,9912	1,9892	0,9604	0,9875	0,9917	0,9933	0,9908	0,9769	0,9897	0,9927	0,9916	0,9933	44178,9
8			Pc = 0,8	1,98488	1,88605	0,98772	0,99064	1,98592	0,98756	0,98821	0,99378	0,97859	0,98472	0,98658	0,97286	0,99238	0,98115	0,98952	43888,1
9			Pc = 0,9	1,9727	1,97959	0,97881	0,9905	1,97739	0,98019	0,97655	0,9522	0,96425	0,97871	0,98919	0,98414	0,9487	0,97883	0,98206	43677,6

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
1	Pop = 100	Pm = 0,05	Pc = 0,1	1,9926	1,9944	0,9931	0,9941	1,9942	0,9933	0,9948	0,9977	0,9952	0,9957	0,9958	0,9949	0,9965	0,9962	0,9957	44481,3
2			Pc = 0,2	1,9988	1,9931	0,9908	0,993	1,9927	0,9959	0,9952	0,9977	0,9956	0,9959	0,9973	0,9966	0,9944	0,9884	0,9989	44487,1
3			Pc = 0,3	1,9969	1,9951	0,9955	0,9932	1,9975	0,9949	0,9953	0,9969	0,9964	0,9964	0,9944	0,9968	0,9919	0,9932	0,9973	44497,7
4			Pc = 0,4	1,9954	1,9955	0,9905	0,9974	1,9951	0,9968	0,9967	0,9942	0,9963	0,9978	0,9929	0,9944	0,9951	0,9067	0,9955	44293,6
5			Pc = 0,5	1,99626	1,99576	0,99467	0,99401	1,99755	0,9977	0,99465	0,99543	0,99451	0,99628	0,99645	0,99153	0,99402	0,99696	0,99651	44495
6			Pc = 0,6	1,99861	1,99065	0,99206	0,99208	1,89449	0,99601	0,99462	0,9961	0,99613	0,9943	0,98965	0,9967	0,99669	0,99533	0,99687	44239,7
7			Pc = 0,7	1,99655	1,99364	0,99043	0,99544	1,99308	0,98961	0,99355	0,99746	0,99309	0,99106	0,99589	0,99139	0,99073	0,99559	0,99488	44421,9
8			Pc = 0,8	1,98699	1,9912	0,9948	0,98748	1,99259	0,98964	0,99322	0,99259	0,99495	0,98973	0,98457	0,99085	0,99426	0,99148	0,99376	44346,5
9			Pc = 0,9	1,9855	1,96977	0,97801	0,98088	1,97652	0,98093	1,04206	0,98338	0,98467	0,99108	0,97462	0,98478	0,98416	0,98458	0,98384	44234,6

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
10	Pop = 30	Pm = AF	Pc = 0,1	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
11		Hybrid	Pc = 0,2	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
12			Pc = 0,3	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
13			Pc = 0,4	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
14			Pc = 0,5	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
15			Pc = 0,6	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
16			Pc = 0,7	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
17			Pc = 0,8	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
18			Pc = 0,9	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
10	Pop = 50	Pm = AF	Pc = 0,1	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
11		Hybrid	Pc = 0,2	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
12			Pc = 0,3	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
13			Pc = 0,4	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
14			Pc = 0,5	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
15			Pc = 0,6	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
16			Pc = 0,7	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
17			Pc = 0,8	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488
18			Pc = 0,9	1,645	2	0	1	2	1	2	1	1	1	1	1	1	1	1	47488

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

[illegible]

LAYER 3

1. Hasil Rata-rata Uji Coba

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit	
1	Pop = 30	Pm = AF	Pc = 0,1	0,8201	0,91	2	2	0,9425	0,9605	1,991	1,9991	0,9584	0,9625	0,9681	0,9993	0,9689	0,969	1	0,9921	50485,4
2			Pc = 0,2	0,7133	0,9069	1,9996	2	0,9579	0,9552	1,9979	1,9968	0,9787	0,9748	0,9663	0,9962	0,9583	0,9751	0,9879	0,9902	50426,7
3			Pc = 0,3	0,7247	0,9209	2	2	0,9864	0,9708	1,9977	1,9957	0,953	0,9375	0,974	1	0,9444	0,9588	0,9979	0,9825	50399,6
4			Pc = 0,4	0,7446	0,9264	2	2	0,9482	0,9817	1,9943	2	0,9752	0,9165	0,9607	0,9985	0,967	0,9616	0,9931	0,9804	50383,9
5			Pc = 0,5	0,7887	0,9214	2	2	0,9564	0,9464	1,9947	1,9744	0,9688	0,9558	0,9632	0,9971	0,955	0,9609	0,9879	0,9864	50343,9
6			Pc = 0,6	0,8014	0,9554	2	2	0,9727	0,9447	1,9799	1,976	0,9388	0,9472	0,9686	0,992	0,9647	0,9378	0,9842	0,9926	50316,1
7			Pc = 0,7	0,8521	0,9292	2	2	0,9521	0,9528	1,9937	1,9843	0,9386	0,9635	0,954	0,9748	0,9675	0,9334	0,9935	0,9829	50304,6
8			Pc = 0,8	0,8848	0,9244	2	1,9997	0,9493	0,9507	1,9752	1,9877	0,9628	0,9569	0,9608	0,9813	0,9577	0,9447	0,9894	0,9579	50280,7
9			Pc = 0,9	0,904	0,9613	1,9918	1,993	0,9342	0,9668	1,9354	1,9786	0,905	0,9451	0,9845	0,9873	0,9595	0,9673	0,9838	0,9794	50178,6

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit	
82	Pop = 50	Pm = AF	Pc = 0,1	0,6337	0,8916	1,9997	2	0,9707	0,9782	1,9989	1,9954	0,9642	0,9725	0,9837	1	0,9769	0,9644	0,9971	0,9984	50459,4
83			Pc = 0,2	0,6524	0,8983	2	2	0,975	0,9692	1,9993	2	0,9621	0,9823	0,9702	0,9994	0,9542	0,9779	0,9982	0,9937	50455,3
84			Pc = 0,3	0,6373	0,9314	2	2	0,9519	0,9791	1,9981	1,9983	0,9832	0,9528	0,9699	0,9992	0,9653	0,9608	0,9996	0,9946	50445,6
85			Pc = 0,4	0,6899	0,9072	2	2	0,9652	0,9585	1,9986	1,9971	0,9502	0,9733	0,978	0,992	0,9764	0,9569	0,9983	0,9945	50419,9
86			Pc = 0,5	0,6901	0,8883	2	2	0,9836	0,9794	1,998	1,9999	0,936	0,9575	0,9841	0,9995	0,9614	0,9671	0,9982	0,9943	50429,2
87			Pc = 0,6	0,7141	0,9207	2	2	0,9707	0,9794	1,9944	1,8977	0,9455	0,9687	0,9633	0,9795	0,9658	0,9663	0,9966	0,9741	50152,8
88			Pc = 0,7	0,7748	0,9332	2	2	0,9333	0,9503	1,9963	1,9919	0,9496	0,9594	0,966	0,9916	0,949	0,9667	0,997	0,9917	50356,5
89			Pc = 0,8	0,8367	0,9459	2	2	0,9777	0,964	1,9936	1,9931	0,9324	0,9292	0,9586	0,9815	0,9512	0,9342	0,9753	0,9899	50302,2
90			Pc = 0,9	0,9315	0,951	1,9991	1,997	0,9573	0,9329	1,9666	1,9745	0,9664	0,9257	0,9531	0,9729	0,9319	0,96	0,988	0,9741	50205,1

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit	
163	Pop = 100 Pm = AF	Pc = 0,1	0,4187	0,9305	2	2	0,9943	0,9923	1,9994	1,9992	0,99	0,9834	0,9941	0,9995	0,9886	0,9873	0,9993	1	50577,1
164		Pc = 0,2	0,4839	0,9109	2	2	0,9862	0,9969	1,999	1,9992	0,9881	0,988	0,9778	0,9996	0,9791	0,9849	0,9987	0,999	50542,8
165		Pc = 0,3	0,5461	0,8944	2	2	0,9963	0,9868	1,9993	1,9988	0,9797	0,9724	0,9839	0,9987	0,9711	0,9813	0,9991	0,9981	50511,1
166		Pc = 0,4	0,4876	0,91	2	2	0,9844	0,9859	1,9999	1,999	0,9841	0,9878	0,979	0,9986	0,9929	0,9883	0,9992	0,9963	50541,9
167		Pc = 0,5	0,5252	0,938	2	2	0,9855	0,9776	1,9991	1,9995	0,9673	0,9731	0,9769	0,999	0,9722	0,9876	1	0,9987	50515,3
168		Pc = 0,6	0,5596	0,9446	2	2	0,9782	0,9838	1,9991	1,9989	0,9734	0,9686	0,976	0,9999	0,9576	0,9669	1	0,9975	50489,2
169		Pc = 0,7	0,7	0,9161	2	2	0,9807	0,9567	1,9989	1,9972	0,9672	0,9683	0,9269	1	0,9598	0,9702	0,9994	0,9959	50416,3
170		Pc = 0,8	0,7814	0,888	2	2	0,9405	0,9677	1,9966	1,9987	0,943	0,9542	0,9569	0,9976	0,974	0,9657	0,9992	0,9948	50377,5
171		Pc = 0,9	0,8683	0,92	2	2	0,9428	0,9584	1,993	1,9825	0,9508	0,9653	0,9629	0,9902	0,9304	0,9542	0,9743	0,9786	50286

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit	
1	Pop = 30	Pm = 0,1	Pc = 0,1	0,9647	0,9674	1,9859	1,9845	0,9764	0,99	0,9598	0,9837	0,9879	0,9854	0,9771	0,9814	0,9837	0,9746	0,97811	0,9782	45984,3
2	Uniform		Pc = 0,2	0,9717	0,9829	1,9851	1,9814	0,9837	0,9843	0,9751	0,9745	0,9826	0,9767	0,9901	0,9898	0,9874	0,9892	0,9824	0,9862	46088,1
3		Pc = 0,3	0,9912	0,9858	1,9871	1,7811	0,9887	0,9875	0,988	0,9817	0,9648	0,983	0,98	0,9807	0,9837	0,9826	0,9738	0,9787	0,9789	44929,9
4		Pc = 0,4	0,9933	0,9887	1,9752	1,9917	1,1836	0,9781	0,98226	0,977	0,9783	0,9847	0,9856	0,9817	0,9743	0,9827	0,9869	0,9827		46448
5		Pc = 0,5	0,9826	0,9948	1,9856	1,9871	0,9869	0,981	0,9803	0,9811	0,9772	0,9696	0,9906	0,9923	0,9894	0,9864	0,9843	0,9789		46151
6		Pc = 0,6	0,9828	0,9826	1,9873	1,9906	0,9779	0,9786	0,9879	0,9828	0,9873	0,9877	0,9654	0,9811	0,9832	0,9838	0,9789	0,9887		46130,6
7		Pc = 0,7	0,9829	0,9882	1,9867	1,9788	0,9819	0,9837	0,9857	0,9849	0,9655	0,9798	0,9677	0,9839	0,985	0,9838	0,9793	0,9841		46038,5
8		Pc = 0,8	0,9901	0,9853	1,9875	1,9661	0,9754	0,973	0,9865	0,9856	0,979	0,9863	0,9811	0,979	0,9702	0,9758	0,9816	0,9851		45958,8
9		Pc = 0,9	0,9669	0,9636	1,9775	1,965	0,9621	0,9662	0,9763	0,974	0,9665	0,9484	0,968	0,9529	0,961	0,9702	0,9806	0,9468		45282,7

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit	
1	Pop = 50	Pm = 0,1	Pc = 0,1	0,9885	0,9845	1,9885	1,9845	0,9837	0,994	0,9928	0,9892	0,9796	0,9845	0,9886	0,986	0,9788	0,9792	0,9763	0,9875	46182,6
2		Uniform	Pc = 0,2	0,9851	0,9915	1,99	1,991	0,9874	0,9917	0,985	0,9875	0,9881	0,9787	0,9865	0,9845	0,981	0,9862	0,9874	0,9934	46266,8
3			Pc = 0,3	0,986	0,9902	1,9938	1,9956	0,9942	0,9886	0,9881	0,9927	0,9793	0,9774	0,9867	0,9861	0,9869	0,9938	0,9906	0,9916	46348,2
4			Pc = 0,4	0,9953	0,9945	1,9928	1,9928	1,1907	0,9836	0,9888	0,9912	0,9956	0,9899	0,9866	0,9835	0,9861	0,9833	0,9803	0,9935	46691,7
5			Pc = 0,5	0,9901	0,9872	1,9938	1,9883	0,9894	0,9964	0,9967	0,9894	0,9893	0,9926	0,991	0,9879	0,9851	0,9828	0,9886	0,9939	46359,4
6			Pc = 0,6	0,9861	0,9833	1,9946	1,9952	0,9918	0,9907	0,9915	0,9869	0,9815	0,9797	0,9836	0,9831	0,9917	0,9924	0,9943	0,9901	46339
7			Pc = 0,7	0,9845	0,9898	1,9922	1,9949	0,9777	0,9738	0,9929	0,9929	0,992	0,9872	0,9844	0,9893	0,9837	0,9816	0,9923	0,9919	46310,6
8			Pc = 0,8	0,9915	0,9947	1,9924	1,9897	0,9832	0,9936	0,9906	0,9768	0,9885	0,9898	0,9826	0,9913	0,9799	0,9808	0,988	0,9948	46287
9			Pc = 0,9	0,9872	0,9831	1,9867	1,9876	0,9829	0,9787	0,9814	0,9801	0,9842	0,9732	0,9644	0,9772	0,9855	0,9903	0,9881	0,9878	46095,6

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit	
1	Pop = 100	Pm = 0,1	Pc = 0,1	0,9945	0,9889	1,9929	1,9931	0,9943	0,9907	0,9936	0,992	0,9913	0,9858	0,9913	0,98	0,9909	0,9868	0,9872	0,9924	46369,4
2		Uniform	Pc = 0,2	0,9858	0,9947	1,9967	1,9943	0,9904	0,9925	0,9975	0,9951	0,9934	0,993	0,9907	0,971	0,9899	0,994	0,9938	0,9951	46434,6
3			Pc = 0,3	0,9913	0,9905	1,9983	1,9938	0,9909	0,9948	0,9923	0,9933	0,9899	0,9935	0,9899	0,9948	0,9916	0,9923	0,9912	0,9934	46464,2
4			Pc = 0,4	0,9933	0,9957	1,9923	1,9946	0,9949	0,9915	0,9945	0,9916	0,9879	0,9919	0,9959	0,99	0,9965	0,9947	0,9948	0,994	46470
5			Pc = 0,5	0,996	0,9921	1,9961	1,9967	0,9922	0,9922	0,9935	0,9932	0,9932	0,9924	0,9907	0,9908	0,9916	0,9953	0,994	0,9942	46487
6			Pc = 0,6	0,9947	0,9946	1,9952	1,9954	0,9961	0,9922	1,0917	0,9963	0,995	0,9926	0,9939	0,9945	0,9911	0,9949	0,9927	0,9954	46734,5
7			Pc = 0,7	0,9955	0,9953	1,998	1,9954	0,9956	0,9945	0,9971	0,9833	0,9934	0,993	0,9961	0,992	0,994	0,9934	0,9926	0,9973	46506,9
8			Pc = 0,8	0,9938	0,9946	1,9955	1,991	0,9864	0,9912	1,0186	1,0917	0,9931	0,9942	0,991	0,9896	0,9976	0,994	0,9892	0,9925	46721,5
9			Pc = 0,9	0,8846	0,9891	1,9921	1,986	0,9859	0,993	0,9887	0,9804	0,9858	0,9745	0,9868	0,99	0,9855	0,9896	0,9922	0,9821	46155

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit	
1	Pop = 30	Pm = 0,05	Pc = 0,1	0,9774	0,988	1,989	1,9886	0,9875	0,9842	0,9937	0,9819	0,9916	0,9851	0,99	0,9777	0,9855	0,9801	0,9851	0,9865	46216,6
2		Uniform	Pc = 0,2	0,981	0,9894	1,9918	1,9934	0,985	0,9873	0,9859	0,98	0,9827	0,9879	0,9808	0,9857	0,9705	0,9762	0,982	0,9854	46188,4
3			Pc = 0,3	0,9901	0,9777	1,9862	1,9944	0,9771	0,9869	0,9815	0,9872	0,9753	0,9833	0,9898	0,9884	0,9804	0,9793	0,9914	0,9894	46201,5
4			Pc = 0,4	0,9871	0,9911	1,9888	1,9878	0,9848	0,9803	0,9903	0,9857	0,9893	0,9836	0,98585	0,9837	0,9808	0,9833	0,9842	0,9904	46216,2
5			Pc = 0,5	0,981	0,98	1,9842	1,986	0,9779	0,9907	0,9784	0,9863	0,9869	0,9796	0,983	0,9847	0,9844	0,9829	0,9732	0,9875	46102,6
6			Pc = 0,6	0,9795	0,9884	1,9871	1,8826	0,9927	0,9817	0,9925	0,9994	0,9896	0,9791	0,9887	0,98446	0,9832	0,9723	0,9535	0,9822	45552,4
7			Pc = 0,7	0,979	0,9721	1,9791	1,9753	0,9735	0,9741	0,983	0,9543	0,98	0,9815	0,9821	0,9792	0,9692	0,9886	0,9805	0,9873,7	
8			Pc = 0,8	0,9852	0,9712	1,9826	1,9803	0,9797	0,9745	0,9917	0,9647	0,9781	0,9548	0,9681	0,9716	0,9848	0,9845	0,9793	0,9728	45876,9
9			Pc = 0,9	0,977	0,9603	1,9534	1,9557	0,9557	0,948	0,9744	0,9499	0,9664	0,9654	0,9714	0,9714	0,9585	0,9617	0,981	0,9649	45317,7

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit	
1	Pop = 50	Pm = 0,05	Pc = 0,1	0,9791	0,9925	1,9914	1,9931	0,9853	0,983	0,9951	0,9896	0,9835	0,9908	0,9825	0,9921	0,9917	0,98	0,9835	0,9864	46299,3
2		Uniform	Pc = 0,2	0,9879	0,9929	1,9894	1,9883	0,9938	0,9872	0,9961	0,9879	0,9877	0,9848	0,988	0,988	0,9919	0,9874	0,9895	0,9942	46332,4
3			Pc = 0,3	0,9942	0,9921	1,9928	1,9938	0,9934	0,9922	0,992	0,9902	0,994	0,9935	0,9906	0,99	0,9953	0,9901	0,993	0,9919	46437,6
4			Pc = 0,4	0,9957	0,9944	1,992	1,9935	1,0933	0,9935	0,9887	0,99	0,9868	0,9882	0,991	0,9929	0,9891	0,9892	0,9908	0,9892	46573,2
5			Pc = 0,5	0,99	0,9946	1,9866	1,9898	0,9927	0,9886	0,9919	0,9912	0,9843	0,9886	0,9927	0,984	0,9883	0,9925	0,9847	0,994	46322,8
6			Pc = 0,6	0,9952	0,9786	1,9947	1,9924	0,9874	0,9929	0,9877	0,9865	0,9913	0,9882	0,9764	0,9917	0,994	0,9952	0,9944	0,9913	46361,3
7			Pc = 0,7	0,9814	0,9864	1,9852	1,9824	0,9903	0,9761	0,9805	0,9873	0,9869	0,9783	0,9713	0,9934	0,9925	0,9908	0,9864	0,9886	46158,1
8			Pc = 0,8	0,982	0,9862	1,9799	1,9795	0,9893	0,9785	0,9737	0,9827	0,9856	0,9882	0,9802	0,9891	0,9837	0,9842	0,9875	0,989	46091,2
9			Pc = 0,9	0,9581	0,9901	1,9752	1,9821	0,9651	0,9748	0,957	0,9801	0,9738	0,9722	0,9727	0,9821	0,9732	0,9745	0,9601	0,9756	45772

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

No	Uji Coba	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit		
1	Pop = 100	Pm = 0,05	Pc = 0,1	0,9961	0,996	1,9948	1,9954	0,9953	0,9943	0,9926	0,9953	0,9946	0,9962	0,993	0,9937	0,99503	0,9929	0,9955	0,994	46517,4
2	Uniform	Pc = 0,2	0,9966	0,9958	1,9981	1,9954	0,9951	0,9951	0,9956	0,9955	0,9956	0,9954	0,9965	0,9945	0,995	0,9926	0,9981	0,9961	0,994	46558,9
3		Pc = 0,3	0,9841	0,9978	1,998	1,9967	0,9973	0,995	0,9925	0,9936	0,9941	0,9949	0,9949	0,9971	0,9916	0,9945	0,9954	0,9957	0,9940	46540,9
4		Pc = 0,4	0,9883	0,9966	1,9948	1,9932	0,9919	0,9927	0,9918	0,9915	0,9973	0,9948	0,9911	0,995	0,9935	0,9926	0,9949	0,9948	0,9948	46478,7
5		Pc = 0,5	0,996	0,9945	1,9954	1,9984	0,9964	0,996	0,9919	0,9965	0,9982	0,9951	0,9788	0,997	0,9924	0,9969	0,9956	0,9981	0,9953	4638,3
6		Pc = 0,6	0,9965	0,993	1,9943	1,9967	0,9953	0,9968	0,9932	0,9963	0,9933	0,9953	0,9959	0,9979	0,9983	0,994	0,9954	0,9961	0,9948	46458,8
7		Pc = 0,7	0,9926	0,9869	1,9926	1,9904	0,9923	0,9945	0,9957	0,9954	0,9918	0,9956	0,9958	0,9971	0,9962	0,9939	0,9919	0,9957	0,9946	46469,2
8		Pc = 0,8	0,9879	0,9944	1,9928	1,9913	0,9928	0,9953	0,9929	0,989	0,9934	0,9952	0,9886	0,9924	0,9928	0,9928	0,9908	0,9942	0,9943	46433,9
9		Pc = 0,9	0,9897	0,9894	1,9892	1,991	0,9836	0,9879	0,9872	0,9901	0,9883	0,9762	0,9917	0,9878	0,9854	0,9867	0,9901	0,9895	0,9880	46280,7

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

[illegible]

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

[illegible]

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit
172	Pop = 100	Pm = AF	Pc = 0,1	0,227	1	2	2	1	1	2	2	1	1	1	1	1	1	1	50664,4
173		Hybrid	Pc = 0,2	0,227	1	2	2	1	1	2	2	1	1	1	1	1	1	1	50664,4
174			Pc = 0,3	0,227	1	2	2	1	1	2	2	1	1	1	1	1	1	1	50664,4
175			Pc = 0,4	0,227	1	2	2	1	1	2	2	1	1	1	1	1	1	1	50664,4
176			Pc = 0,5	0,227	1	2	2	1	1	2	2	1	1	1	1	1	1	1	50664,4
177			Pc = 0,6	0,227	1	2	2	1	1	2	2	1	1	1	1	1	1	1	50664,4
178			Pc = 0,7	0,227	1	2	2	1	1	2	2	1	1	1	1	1	1	1	50664,4
179			Pc = 0,8	0,227	1	2	2	1	1	2	2	1	1	1	1	1	1	1	50664,4
180			Pc = 0,9	0,227	1	2	2	1	1	2	2	1	1	1	1	1	1	1	50664,4

LAYER 4

1. Hasil Rata-rata Uji Coba

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit
1	Pop = 30	Pm = AF	Pc = 0,1	0,8345	1	0,9866	1	0,9981	0,7434	0,9868	0,9692	0,9921	1,9818	1,9969	0,9873	0,9904	1	1,8999	1	43899,44
2			Pc = 0,2	0,9106	0,9995	0,9857	1	0,9952	0,6921	0,9686	0,9682	0,9921	1,9843	1,9968	0,9829	0,9822	0,9999	2	1	44489,31
3			Pc = 0,3	0,9096	1	0,9739	1	0,9952	0,7766	0,9664	0,978	0,9779	1,9771	1,996	0,9748	0,9494	1	2	1	44426,54
4			Pc = 0,4	0,9007	0,9999	0,9617	1	0,9927	0,7333	0,982	0,9783	0,9753	1,9808	1,9917	0,9849	0,9835	0,9999	2	1	44462,95
5			Pc = 0,5	0,9193	0,9981	0,9665	1	0,9939	0,7611	0,9817	0,9758	0,9888	1,9491	1,9905	0,9831	0,9655	0,9999	2	1	44434,96
6			Pc = 0,6	0,8786	0,9992	0,9859	1	0,9857	0,7869	0,9811	0,981	0,9785	1,9707	1,9878	0,9642	0,976	1	2	0,9996	44428,12
7			Pc = 0,7	0,9082	1	0,9742	0,9995	0,9932	0,8096	0,9624	0,9667	0,9592	1,9585	1,9952	0,9771	0,9733	1	2	0,9998	44398,63
8			Pc = 0,8	0,9041	1	0,9862	0,9993	0,991	0,8585	0,9012	0,9763	0,9752	1,9722	1,9897	0,9566	0,9791	0,9997	2	1	44363,81
9			Pc = 0,9	0,9652	0,9888	0,927	0,9917	0,9792	0,9166	0,9835	0,9604	0,977	1,9482	1,959	0,9599	0,959	0,9978	1,9993	0,9956	44244,96

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit
82	Pop = 50	Pm = AF	Pc = 0,1	0,9094	0,9998	0,993	0,9991	1	0,5792	0,9978	0,9959	0,9929	1,9843	1,9994	0,9949	0,9952	1	2	1	44581,78
83			Pc = 0,2	0,8734	0,9992	0,9958	1	0,9984	0,6533	0,9899	0,9915	0,9954	1,9903	1,9983	0,9709	0,9958	0,9997	2	1	44539,07
84			Pc = 0,3	0,9122	0,9995	0,9851	1	0,9985	0,6289	0,9914	0,9962	0,9929	1,9887	1,998	0,9795	0,9805	0,9999	2	1	44542,6
85			Pc = 0,4	0,8646	0,9994	0,981	1	0,9978	0,6981	0,9888	0,9901	0,9893	1,9853	1,9975	0,9837	0,9833	1	1,9999	1	44510,12
86			Pc = 0,5	0,8518	0,999	0,9809	1	0,9917	0,7298	0,9881	0,983	0,9909	1,9871	1,9978	0,9857	0,9781	1	2	1	44490,66
87			Pc = 0,6	0,8901	0,9996	0,9796	1	0,9966	0,723	0,9816	0,992	0,9855	1,9819	1,9969	0,9666	0,9807	1	2	1	44492,19
88			Pc = 0,7	0,8819	0,9984	0,9767	1	0,9917	0,8045	0,9793	0,9803	0,9741	1,9675	1,9929	0,9597	0,9726	1	1,9999	1	44417,48
89			Pc = 0,8	0,891	0,9971	0,9896	1	0,9944	0,8066	0,9605	0,9507	0,9712	1,9775	1,981	0,9706	0,9864	1	2	1	44404,08
90			Pc = 0,9	0,9381	0,9972	0,9549	0,9977	0,9932	0,8171	0,9749	0,9435	0,9662	1,9493	1,70729	0,9787	0,9833	0,9979	1,9997	0,9994	43868,74

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit
163	Pop = 100 Pm = Af	Pc = 0,1	0,9067	1	0,9947	0,9998	0,9995	0,55	0,9981	0,9983	0,9962	1,9979	2	0,9984	0,996	1	2	1	44606,02
164		Pc = 0,2	0,9335	0,9995	0,9968	0,9996	0,9985	0,5362	0,9982	0,998	0,9981	1,991	2	0,9917	0,9992	0,9996	2	1	44609,46
165		Pc = 0,3	0,8892	0,9993	0,998	1	0,999	0,5738	0,9966	0,996	0,997	1,996	1,9989	0,9981	0,9966	1	2	1	44594,88
166		Pc = 0,4	0,8864	0,9991	0,996	1	0,9989	0,5778	0,9964	0,9971	0,9971	1,9976	2	0,9962	0,9976	1	2	1	44594,86
167		Pc = 0,5	0,9048	0,9995	0,9947	0,9998	0,9981	0,5868	0,998	0,9915	0,9948	1,9869	1,9988	0,9933	0,9934	1	2	1	44575,53
168		Pc = 0,6	0,9152	1	0,9931	1	0,9996	0,5946	0,9965	0,9923	0,9717	1,9962	1,9982	0,9849	0,9946	0,9996	2	1	44554,99
169		Pc = 0,7	0,8756	1	0,9899	0,9989	0,9985	0,6401	0,995	0,99	0,9913	1,9954	1,9978	0,9944	0,9846	0,9999	2	0,9997	44549,91
170		Pc = 0,8	0,9251	0,9995	0,9905	0,9998	0,9983	0,6463	0,9923	0,9887	0,9871	1,9702	1,8943	0,9926	0,9652	0,9994	2	1	44336,21
171		Pc = 0,9	0,8903	0,9991	0,9895	0,9992	0,9901	0,7631	0,9792	0,9881	0,9705	1,9667	1,9867	0,9791	0,9681	1	2	1	44438,85

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit
1	Pop = 30 Uniform	Pc = 0,1	0.9835	0.9767	0.98	0.96382	0.9782	0.9738	0.9767	0.9819	0.9873	0.9841	0.9809	0.9802	0.9803	0.9849	0.9797	0.989	35088
2		Pc = 0,2	0.987	0.9837	0.9756	0.9848	0.9833	0.9878	0.9782	0.9861	0.9793	0.97947	0.9669	0.9676	0.9761	0.9803	0.987	0.9857	35136.52
3		Pc = 0,3	0.9836	0.9851	0.9748	0.9836	0.9819	0.9715	0.99	0.9812	0.9832	0.9944	0.9754	0.9833	0.9794	0.9895	0.9865	0.9884	35252.68
4		Pc = 0,4	0.9625	0.9852	0.9888	0.9881	0.983	0.9589	0.9876	0.9838	0.9792	0.9726	0.9824	0.9792	0.9841	0.9883	0.9887	0.9941	35272.15
5		Pc = 0,5	0.9816	0.9898	0.9791	0.98	0.9818	0.9656	0.9796	0.9781	0.9825	1.0696	0.9815	0.9811	0.9844	0.9825	1.0252	0.9936	35577.49
6		Pc = 0,6	0.9753	0.9846	0.9911	0.9836	0.9772	0.9765	0.9778	0.9819	0.9832	0.9832	0.9707	0.9912	0.9867	0.9829	1.0548	0.9824	35596.45
7		Pc = 0,7	0.9812	0.9917	0.9843	0.9732	0.9762	0.9714	0.9909	0.9846	0.9767	0.9809	0.9784	0.9828	0.9833	0.9772	0.9797	0.983	35108.18
8		Pc = 0,8	0.9815	0.9798	0.982	0.9758	0.977	0.9522	0.9813	0.9772	0.9859	0.9705	0.9829	0.9703	0.9667	0.9726	0.9745	0.9757	34929
9		Pc = 0,9	0.9708	0.9556	0.9715	0.9589	0.9622	0.9663	0.962	0.9606	0.9674	0.973	0.954	0.96	0.9842	0.9805	0.9865	0.9683	34725.01

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit
1	Pop = 50 Uniform	Pc = 0,1	0,9884	0,98259	0,98865	0,98626	0,9821	0,97784	0,9909	0,98598	0,98906	0,98547	0,98045	0,98563	0,98278	0,99127	0,98797	0,98941	35325,85
2		Pc = 0,2	0,98682	0,99286	0,98684	0,98816	0,9888	0,98605	0,98387	0,98493	0,98793	0,98123	0,98323	0,9839	0,98799	0,99125	0,99191	0,98593	35369,41
3		Pc = 0,3	0,98757	0,98686	0,99002	0,98605	0,98771	0,99137	0,98954	0,9757	0,98883	0,98798	0,9909	0,98383	0,98504	0,99343	0,99281	0,99044	35401,16
4		Pc = 0,4	0,99093	0,99204	0,99313	0,99544	0,98267	0,98582	0,98473	0,98479	0,98583	0,98881	0,99068	0,98531	0,99088	0,98587	1,09574	0,99097	36037,67
5		Pc = 0,5	0,98619	0,99402	0,9859	0,98991	0,98619	0,97497	0,98637	0,98273	0,98571	0,99438	0,99064	0,99328	0,98824	0,98779	0,98895	0,9956	35418,67
6		Pc = 0,6	0,99305	0,99046	0,98551	0,98865	0,98914	0,97747	0,99029	0,98199	0,99125	0,98698	0,98835	0,98793	0,99152	0,9943	1,09549	0,98769	36041,72
7		Pc = 0,7	0,98724	0,99119	0,99213	0,98799	0,99357	0,98532	0,99082	0,98608	0,99177	1,09312	0,98229	0,98195	0,98913	0,99227	0,9939	0,99438	35606,99
8		Pc = 0,8	0,98071	0,99002	0,98713	0,99534	0,98341	0,97139	0,97809	0,97879	0,98169	0,99076	0,98722	0,99026	0,98399	0,99298	0,99281	0,98703	35379,07
9		Pc = 0,9	0,97463	0,97539	0,97665	0,98273	0,98583	0,97941	0,98036	0,9682	0,97717	0,97143	0,97858	0,9746	0,97948	0,98204	0,98784	0,9898	35127,6

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit	
1	Pop = 100	Pm = 0,1	Pc = 0,1	0,98706	0,99431	0,99488	0,99102	0,99466	0,9815	0,99048	0,99336	0,98846	0,98892	0,99264	0,99382	0,98734	0,99366	0,99298	0,98893	35497,34
2		Uniform	Pc = 0,2	0,99557	0,99234	0,99189	0,98953	0,98405	0,98466	0,99349	0,99371	0,99186	0,98734	0,99122	0,99409	0,99238	0,99467	0,9961	0,99654	35542,75
3			Pc = 0,3	0,99535	0,99495	0,99121	0,99238	0,99195	0,99206	0,99224	0,99135	0,99104	0,98977	0,99221	0,99158	0,99489	0,99566	0,99567	0,99429	35565,93
4			Pc = 0,4	0,9941	0,99426	0,99282	0,99578	0,99384	0,99471	0,99494	0,99499	0,99404	0,98942	0,99378	0,99253	0,98706	0,99509	0,99722	0,99801	35608,51
5			Pc = 0,5	0,99476	0,99445	0,99507	0,99441	0,99285	0,9928	0,99242	0,99363	0,99353	0,99504	0,99246	0,99487	0,99595	0,99379	0,99796	0,99645	35615,02
6			Pc = 0,6	0,99374	0,99443	0,99149	0,9918	0,99703	0,995	0,99523	0,99286	0,992	0,99237	0,99723	0,99315	0,99378	0,99299	0,99364	0,99328	35565,27
7			Pc = 0,7	0,99312	0,99345	0,99568	0,99447	0,99049	0,99321	0,99125	0,987	0,99404	0,99391	0,99592	0,9964	0,99698	0,99469	0,99529	0,9951	35588,14
8			Pc = 0,8	0,98926	0,99517	0,9901	0,98883	0,99123	0,99045	0,99322	0,98972	0,99379	0,98947	0,99065	0,99156	0,98981	0,9936	0,99497	0,99629	35532,41
9			Pc = 0,9	0,9899	0,98991	0,98856	0,99032	0,98862	0,98685	0,99042	0,9884	0,98533	0,98578	0,98227	0,99338	0,98947	0,99355	1,09109	0,9925	36043,28

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit	
1	Pop = 30	Pm = 0,05	Pc = 0,1	0,9759	0,9895	0,9871	0,983	0,9869	0,9818	0,9792	0,9834	0,9812	0,9857	0,986	0,9859	0,98	0,9896	0,9894	0,984	35280,8
2	Uniform		Pc = 0,2	0,9816	0,9864	0,987	0,9913	0,9887	0,9788	0,9893	0,9914	0,9739	0,9848	0,9794	0,9852	0,9839	0,9829	0,9919	0,9893	35323,37
3			Pc = 0,3	0,9858	0,98281	0,9879	0,9834	0,987	0,981	0,9863	0,989	0,9824	0,9853	0,9881	0,9889	0,9883	0,9919	0,9991	0,9873	35398,97
4			Pc = 0,4	0,987	0,9885	0,9808	0,9872	0,984	0,986	0,9844	0,9793	0,978	0,9868	0,9811	0,9833	0,9891	0,9906	1,0482	0,9906	35672,57
5			Pc = 0,5	0,9799	0,978	0,9846	0,9843	0,9875	0,9872	0,9862	0,9863	0,9777	0,9886	0,9815	0,9836	0,9915	0,99081	0,9941	0,9865	35323,56
6			Pc = 0,6	0,9844	0,9714	0,9787	0,9867	0,9872	0,981	0,9852	0,9832	0,9852	0,9806	0,9758	0,9774	0,9855	0,98313	1,015	0,9877	35366,96
7			Pc = 0,7	0,9827	0,9821	0,9835	0,9841	0,9781	0,982	0,9844	0,9744	0,9773	0,9767	0,9724	0,9893	0,9795	0,9836	0,9815	0,985	35138,89
8			Pc = 0,8	0,9681	0,9771	0,9787	0,9788	0,9815	0,9757	0,9704	0,9696	0,9727	0,9676	0,99	0,9815	0,9704	0,9674	0,9875	0,9778	34982,22
9			Pc = 0,9	0,9658	0,9675	0,9617	0,9626	0,9685	0,9618	0,9759	0,9524	0,9705	0,9651	0,967	0,9607	0,9792	0,9659	0,9818	0,9664	34684,54

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit	
1	Pop = 50	Pm = 0,05	Pc = 0,1	0,98901	0,99383	0,9923	0,99029	0,99283	0,97827	0,99203	0,99124	0,9913	0,99025	0,98887	0,98891	0,99568	0,99246	0,99239	0,99651	35514,29
2		Uniform	Pc = 0,2	0,99051	0,98642	0,98676	0,99142	0,98808	0,98672	0,99073	0,99206	0,98376	0,98595	0,99301	0,99066	0,99136	0,99267	0,99359	0,99655	35481,56
3			Pc = 0,3	0,99266	0,99252	0,98967	0,99016	0,98382	0,98691	0,99139	0,98797	0,98616	0,98845	0,98627	0,98667	0,98929	0,99244	0,99273	0,99483	35459,03
4			Pc = 0,4	0,98823	0,98995	0,99091	0,98917	0,98994	0,98934	0,99261	0,98357	0,99039	0,98403	0,99304	0,98475	0,98668	0,98853	0,988	0,99144	35399,25
5			Pc = 0,5	0,99562	0,99208	0,99308	0,9904	0,99013	0,97967	0,99173	0,98718	0,99041	0,99524	0,98754	0,99247	0,99526	0,9938	0,99373	0,99181	35509,56
6			Pc = 0,6	0,99486	0,98598	0,98487	0,98653	0,99292	0,98691	0,99284	0,98786	0,99214	0,98601	0,98886	0,98834	0,99035	0,98913	0,98961	0,98878	35399,34
7			Pc = 0,7	0,99025	0,99323	0,98568	0,99012	0,99263	0,98662	0,98697	0,98539	0,98527	0,98875	0,97994	0,99025	0,99082	0,98002	0,98825	0,99334	35366,72
8			Pc = 0,8	0,98465	0,98348	0,98644	0,99344	0,99045	0,974	0,98645	0,98273	0,97498	0,97835	0,99227	0,98931	0,97922	0,98985	0,98502	0,99126	35320,32
9			Pc = 0,9	0,98123	0,9791	0,97337	0,96978	0,9773	0,96737	0,96974	0,97563	0,94812	0,97763	0,98413	0,97359	0,97557	0,97403	0,97514	0,98227	34896,22

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit	
1	Pop = 100	Pm = 0,05	Pc = 0,1	0,99376	0,99576	0,99692	0,99275	0,99496	0,9912	0,99599	0,99605	0,99438	0,99575	0,99317	0,99389	0,99403	0,99408	0,99287	0,99426	35587,15	
2	Uniform			Pc = 0,2	0,9942	0,99559	0,99627	0,99663	0,9948	0,99411	0,99413	0,9915	0,99641	0,99632	0,9941	0,99509	0,99432	0,99613	0,99674	0,99783	35645,35
3				Pc = 0,3	0,9935	0,99522	0,99602	0,9943	0,99307	0,99411	0,99399	0,99577	0,99554	0,99446	0,99721	0,99566	0,99602	0,99646	0,99636	0,99644	35637,29
4				Pc = 0,4	0,99472	0,99351	0,99608	0,99429	0,99631	0,99287	0,99709	0,99382	0,99362	0,99654	0,99036	0,99186	0,99463	0,99654	0,99795	0,99704	35632,94
5				Pc = 0,5	0,99667	0,99578	0,99563	0,99487	0,9932	0,99142	0,99442	0,99406	0,99206	0,99499	0,99595	0,99134	0,99573	0,99448	0,99588	0,99478	35607,65
6				Pc = 0,6	0,9963	0,9949	0,99156	0,99485	0,99189	0,99277	0,99362	0,99524	0,99463	1,09603	0,99298	0,99692	0,9937	0,99595	0,99391	0,99696	35750,59
7				Pc = 0,7	0,98996	0,99205	0,99695	0,99266	0,98993	0,99349	0,9915	0,99369	0,98699	0,99092	0,99321	0,99391	0,99107	0,99459	0,99308	0,99635	35545,63
8				Pc = 0,8	0,99215	0,98665	0,99609	0,99241	0,99135	0,98646	0,99366	0,99232	0,99396	0,98605	0,99634	0,99382	0,99229	0,98952	0,99431	0,99334	35515,5
9				Pc = 0,9	0,97654	0,986	0,98067	0,99316	0,99075	0,98586	0,98627	0,98915	0,98038	0,98023	0,99212	0,98165	0,99117	0,99493	0,99178	0,98116	35362,65

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit
10	Pop = 30	Pm = AF	Pc = 0,1	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
11		Hybrid	Pc = 0,2	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
12			Pc = 0,3	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
13			Pc = 0,4	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
14			Pc = 0,5	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
15			Pc = 0,6	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
16			Pc = 0,7	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
17			Pc = 0,8	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
18			Pc = 0,9	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit
91	Pop = 50	Pm = AF	Pc = 0,1	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
92		Hybrid	Pc = 0,2	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
93			Pc = 0,3	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
94			Pc = 0,4	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
95			Pc = 0,5	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
96			Pc = 0,6	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
97			Pc = 0,7	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
98			Pc = 0,8	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
99			Pc = 0,9	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	Total Profit
172	Pop = 100	Pm = AF	Pc = 0,1	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
173		Hybrid	Pc = 0,2	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
174			Pc = 0,3	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
175			Pc = 0,4	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
176			Pc = 0,5	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
177			Pc = 0,6	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
178			Pc = 0,7	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
179			Pc = 0,8	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08
180			Pc = 0,9	1	1	1	1	1	0,417	1	1	1	2	2	1	1	1	2	1	44663,08

LAYER 5

1. Hasil Rata-rata Uji Coba

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
1	Pop = 30	Pm = AF	Pc = 0,1	0,9242	0,8675	0,9921	0,8003	0,9929	0,9947	1,7584	0,9594	0,9622	1,9849	1,9953	0,9989	1	0,9855	0,9999	54983,8
2			Pc = 0,2	0,9516	0,875	0,9959	0,7794	0,9935	0,9944	1,8002	0,9726	0,9573	1,9986	1,9468	0,9949	1	0,9679	1	54962,2
3			Pc = 0,3	0,9095	0,8843	0,9885	0,8182	0,9947	0,997	1,8181	0,9417	0,9529	1,9857	1,9907	0,9961	1	0,9572	1	54936,9
4			Pc = 0,4	0,9143	0,9024	0,9889	0,7885	0,973	0,9937	1,8437	0,9541	0,9729	1,9834	1,965	0,999	1	0,953	1	54902,5
5			Pc = 0,5	0,9345	0,8872	0,9905	0,8977	0,9931	0,9705	1,8235	0,91	0,9401	1,981	1,8731	0,986	1	0,972	1	54543,3
6			Pc = 0,6	0,9627	0,9197	0,9696	0,8648	0,98	0,95	1,8299	0,923	0,959	1,9663	1,9731	0,9898	1	0,9605	1	54815,1
7			Pc = 0,7	0,9524	0,9092	0,9867	0,8754	0,9805	0,9736	1,7497	0,9712	0,9471	1,9769	1,9721	0,987	1	0,9514	1	54880,2
8			Pc = 0,8	0,954	0,9318	0,9592	0,9163	0,9578	0,9724	1,8277	0,9493	0,9432	1,9497	1,9639	0,9801	0,993	0,9631	0,9945	54722,2
9			Pc = 0,9	0,9846	0,9539	0,9385	0,9573	0,9755	0,9679	1,8335	0,9638	0,9175	1,9257	1,9573	0,969	0,9677	0,9671	0,9948	54598

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
1	Pop = 50	Pm = AF	Pc = 0,1	0,9176	0,8819	0,9984	0,7251	0,9989	0,9961	1,7579	0,9679	0,9764	1,9984	2	0,99773	1	0,9888	0,9997	55044,3
2			Pc = 0,2	0,9324	0,8277	0,9984	0,7471	0,9978	0,998	1,7785	0,965	0,9879	2	1,9952	0,9994	1	0,9825	1	55037,3
3			Pc = 0,3	0,9023	0,8885	0,994	0,7408	0,9898	1	1,7632	0,967	0,9784	1,9966	1,9971	0,998	1	0,99	1	55020,3
4			Pc = 0,4	0,9241	0,8346	0,9978	0,7516	0,9981	0,9961	1,8142	0,9852	0,9387	1,9902	1,9985	0,9996	1	0,9882	1	55007
5			Pc = 0,5	0,931	0,8875	0,9951	0,7405	0,9946	0,9985	1,7453	0,9704	0,9621	1,9947	1,9942	0,9997	1	0,991	1	55027,7
6			Pc = 0,6	0,9705	0,8708	0,9889	0,7937	0,992	0,9845	1,7735	0,9894	0,968	1,9919	1,9799	0,9979	1	0,9274	1	54973,8
7			Pc = 0,7	0,9347	0,9089	0,9909	0,8176	0,9653	0,9949	1,7785	0,959	0,9662	1,9857	1,9826	0,9907	1	0,9526	0,999	54914,1
8			Pc = 0,8	0,9375	0,9163	0,9693	0,8964	0,9813	0,99	1,8351	0,9593	0,9376	1,9687	1,9364	0,9903	1	0,937	0,9983	54797
9			Pc = 0,9	0,9741	0,9334	0,971	0,9377	0,9772	0,9723	1,8275	0,9203	0,9586	1,9606	1,9674	0,9436	0,9835	0,9495	0,989	54666

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit	
1	Pop = 100	Pm = AF	Pc = 0,1	0,9675	0,8878	0,9972	0,7162	0,9991	0,9952	1,6684	0,9937	0,9853	1,9922	2	0,9922	1	0,9937	1	55081,4
2			Pc = 0,2	0,9809	0,8359	0,9993	0,7607	0,9991	1	1,64	0,9921	0,9955	1,9991	1,998	0,9983	1	0,9899	1	55098,5
3			Pc = 0,3	0,948	0,9082	0,9989	0,7111	0,999	0,9992	1,6591	0,9944	0,9805	1,9929	2	1	1	0,9943	1	55093,1
4			Pc = 0,4	0,9792	0,8694	0,9986	0,716	0,9986	0,9971	1,6317	0,9964	0,9974	2	1,9988	0,9995	1	0,9983	1	55118,2
5			Pc = 0,5	0,9449	0,8013	0,9988	0,7504	0,9987	0,9982	1,771	0,9833	0,9813	1,999	1,9966	0,9997	1	0,9834	1	55044,8
6			Pc = 0,6	0,9133	0,87397	0,9987	0,7744	0,9969	0,9082	1,6797	0,9905	0,986	1,9966	1,9965	0,999	1	0,9936	1	54767,1
7			Pc = 0,7	0,9397	0,9007	0,998	0,7543	0,9966	0,9964	1,6855	0,9847	0,9671	1,9987	1,9986	0,998	1	0,97891	1	55050,2
8			Pc = 0,8	0,9284	0,8892	0,9911	0,786	0,991	0,9963	1,7205	0,9793	0,9934	1,9895	1,992	0,9974	1	0,9633	1	55030,8
9			Pc = 0,9	0,9584	0,9107	0,9712	0,8821	0,9928	0,9855	1,8385	0,9417	0,9138	1,9312	1,9781	0,9783	0,9976	0,9693	1	54787,9

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit	
1	Pop = 30	Pm = 0,1	Pc = 0,1	0,97425	0,97658	0,98763	0,96354	0,97976	0,96911	0,97391	0,97446	0,98516	0,98127	0,98591	0,97654	0,98186	0,97935	0,97082	47260,8	
2	Uniform			Pc = 0,2	0,9804	0,97117	0,98706	0,97301	0,97909	0,98456	0,98443	0,97141	0,97041	0,96385	0,96933	0,97264	0,98634	0,96797	0,96109	47107,9
3			Pc = 0,3	0,98032	0,97319	0,97971	0,97689	0,96491	0,97804	0,95669	0,96895	0,97842	0,97476	0,9742	0,95524	0,97105	0,98284	0,97543	47014,9	
4			Pc = 0,4	0,9709	0,97956	0,97704	0,94839	0,9609	0,98139	0,96675	0,95935	0,98466	0,97565	0,97587	0,98588	0,97267	0,97184	0,97357	47038,4	
5			Pc = 0,5	0,95847	0,97188	0,97299	0,96205	0,96851	0,98126	0,96717	0,96923	0,98066	0,95191	0,98435	0,9659	0,96344	0,96515	0,9704	46830,1	
6			Pc = 0,6	0,97614	0,96786	0,96533	0,95208	0,97813	0,96702	0,92292	0,95481	0,95366	0,96202	0,96309	0,95072	0,98034	0,9613	0,96386	46569,8	
7			Pc = 0,7	0,97824	0,95915	0,94179	0,9344	0,94568	0,93832	0,94182	0,93802	0,93175	0,95601	0,93689	0,94221	0,97072	0,94289	0,97232	45961,7	
8			Pc = 0,8	0,94518	0,9401	0,92453	0,96407	0,94396	0,96385	0,95439	0,939	0,92718	0,9455	0,90256	0,94139	0,9684	0,94675	0,92402	45512,3	
9			Pc = 0,9	0,92668	0,88336	0,8326	0,87196	0,92995	0,91681	0,88666	0,93827	0,91138	0,87224	0,86493	0,93122	0,93402	0,82245	0,92439	43531,6	

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit	
1	Pop = 50	Pm = 0,1	Pc = 0,1	0,9867	0,98489	0,98705	0,98404	0,98487	0,98593	0,9796	0,99109	0,97526	0,98054	0,98742	0,98043	0,99244	0,99006	0,98774	47646,7
2		Uniform	Pc = 0,2	0,98589	0,98854	0,99025	0,9824	0,98821	0,98547	0,98753	0,98403	0,99336	0,98999	0,9915	0,98849	0,99086	0,99263	0,99044	47802,6
3			Pc = 0,3	0,99001	0,98707	0,98444	0,98035	0,99051	0,9922	0,99153	0,98654	0,99003	0,98607	0,99363	0,9905	0,98727	0,98937	0,99373	47809,2
4			Pc = 0,4	0,99472	0,98962	0,9894	0,98955	0,9913	0,9921	0,98503	0,98638	0,99151	0,99431	0,98869	0,98957	0,99484	0,98454	0,99001	47871,4
5			Pc = 0,5	0,98908	0,99014	0,99104	0,98926	0,99246	0,99445	0,988	0,98771	0,988	0,98957	0,99163	0,99099	0,99068	0,99277	0,98654	47855,1
6			Pc = 0,6	0,99393	0,99109	0,99216	0,98906	0,98915	0,9892	0,99106	0,99102	0,99224	0,99249	0,98818	0,98917	0,98809	0,98648	0,99456	47872,1
7			Pc = 0,7	0,99217	0,9908	0,98734	0,99158	0,9917	0,98623	0,98229	0,98316	0,99272	1,0908	0,99139	0,99075	0,99336	0,98604	0,99205	48173,2
8			Pc = 0,8	0,99351	0,98734	0,99386	0,98021	0,99006	0,98522	0,97982	0,98856	0,97986	0,98756	0,98505	0,97608	0,98744	0,98695	0,99192	47689,4
9			Pc = 0,9	0,9738	0,9823	0,97378	0,97629	0,98359	0,98063	0,98169	0,97617	0,97914	0,98475	0,98976	0,98597	0,98913	0,98847	0,98831	47521,5

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,1$

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit	
1	Pop = 100	Pm = 0,1	Pc = 0,1	0,99493	0,9923	0,99412	0,99319	0,9897	0,99093	0,99048	0,99298	0,9935	0,99188	0,99183	0,99558	0,99175	0,99184	0,9959	47984,5
2		Uniform	Pc = 0,2	0,9958	0,99669	0,9952	0,99268	0,99226	0,99121	0,9916	0,99441	0,99164	0,99441	0,99477	0,99603	0,9942	0,98969	0,99178	48015,2
3			Pc = 0,3	0,99502	0,99504	0,99415	0,98912	0,99167	0,99174	0,99645	0,99311	0,99429	0,99418	0,99638	0,99327	0,99339	0,99488	0,99727	48046,4
4			Pc = 0,4	0,99491	0,99807	0,99654	0,99311	0,99446	0,99349	0,99194	0,99447	0,99385	0,99626	0,99626	0,99665	0,9938	0,99481	0,99739	48099,2
5			Pc = 0,5	0,99621	0,99292	0,9951	0,99606	0,99607	0,99335	0,98964	0,99147	0,99194	0,99515	0,99641	0,99326	0,9967	0,99457	0,99606	48071,3
6			Pc = 0,6	0,99193	0,99515	0,99437	0,99403	0,99341	0,99298	0,99566	0,99467	0,99601	0,99406	0,99329	0,99507	0,99473	0,99666	0,99716	48075
7			Pc = 0,7	0,99726	0,99249	0,99559	0,99512	0,99566	0,99141	0,99398	0,99552	0,99181	0,99706	0,99288	0,9945	0,99462	0,99613	0,99613	48076,3
8			Pc = 0,8	0,99251	0,99407	0,99178	0,99496	0,99151	0,99132	0,99398	0,99291	0,98856	0,99581	0,98583	0,99398	0,99335	0,99665	0,99163	47965,2
9			Pc = 0,9	0,99375	0,9926	0,99068	0,98566	0,9917	0,99083	0,99059	0,98476	0,98896	1,09152	0,99158	0,99428	0,99158	0,98706	0,99119	48209

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
1	Pop = 30	Pm = 0,05	Pc = 0,1	0,99162	0,98722	0,98575	0,98679	0,99052	0,98644	0,98894	0,98632	0,98355	0,98648	0,97798	0,98279	0,98719	0,98307	0,99373	47688,7
2			Pc = 0,2	0,99083	0,98093	0,9872	0,98319	0,98391	0,98635	0,98235	0,97556	0,98962	0,98962	0,98197	0,99037	0,98938	0,98672	0,98797	47667,5
3			Pc = 0,3	0,98787	0,98403	0,98272	0,99453	0,98935	0,98824	0,98685	0,98517	0,98522	0,98502	0,99068	0,98813	0,99213	0,98629	0,99179	47760
4			Pc = 0,4	0,98903	0,97807	0,98299	0,98545	0,98784	0,99163	0,98263	0,98626	0,9851	0,98748	0,9848	0,9894	0,98743	0,98925	0,99328	47718,6
5			Pc = 0,5	0,98863	0,99209	0,98834	0,97823	0,98244	0,9837	0,97621	0,98349	0,98664	0,9823	0,99007	0,98512	0,98139	0,98962	0,99211	47640,5
6			Pc = 0,6	0,9802	0,98719	0,98611	0,98268	0,98894	0,97956	0,97704	0,99069	0,98649	0,97278	0,98579	0,99056	0,9748	0,9851	0,9899	47548,9
7			Pc = 0,7	0,98371	0,97864	0,9758	0,97978	0,98994	0,97778	0,98466	0,97525	0,97509	0,98366	0,98654	0,97461	0,98306	0,9863	0,987	47448,6
8			Pc = 0,8	0,98777	0,9826	0,97573	0,98638	0,96468	0,97594	0,96105	0,98703	0,97516	0,97706	0,97809	0,97434	0,97288	0,98019	0,9805	47220,4
9			Pc = 0,9	0,95351	0,97842	0,97024	0,95676	0,95215	0,9669	0,96302	0,97554	0,97079	0,98616	0,98329	0,96402	0,96305	0,9687	0,95991	46750,8

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
1	Pop = 50	Pm = 0,05	Pc = 0,1	0,99101	0,99075	0,98943	0,99176	0,99282	0,9867	0,99304	0,99087	0,98786	0,98853	0,99132	0,9918	0,99321	0,9899	0,99081	47880,9
2			Pc = 0,2	0,98905	0,99163	0,99431	0,99181	0,99179	0,99238	0,98555	0,9868	0,99481	0,99081	0,99291	0,99127	0,99403	0,98784	0,99284	47925,2
3			Pc = 0,3	0,98984	0,98927	0,98752	0,99027	0,99318	0,99406	0,99242	0,99182	0,98916	0,98949	0,99261	0,99345	0,99181	0,98666	0,99109	47893,5
4			Pc = 0,4	0,99356	0,98944	0,98886	0,99222	0,99417	0,99283	0,98633	0,9905	0,9891	0,99455	0,99168	0,98992	0,99468	0,99273	0,99398	47946,7
5			Pc = 0,5	0,98957	0,9931	0,99051	0,98911	0,98629	0,99126	0,99607	0,98814	0,99214	0,99159	0,993	0,99266	0,99453	0,99494	0,99335	47940
6			Pc = 0,6	0,99173	0,99053	0,99198	0,98693	0,99004	0,99065	0,99034	0,99122	0,98837	0,99215	0,99067	0,99054	0,99584	0,99334	0,99272	47924,9
7			Pc = 0,7	0,99333	0,9906	0,98475	0,98805	0,98895	0,99066	0,98173	0,99105	0,98071	0,99182	0,98934	0,98706	0,98912	0,98695	0,99165	47783,5
8			Pc = 0,8	0,9859	0,98956	0,98151	0,98666	0,9907	0,9896	0,96267	0,98882	0,98889	0,98384	0,98386	0,98761	0,98344	0,97757	0,98244	47583,5
9			Pc = 0,9	0,97748	0,97642	0,97204	0,95802	0,97399	0,9739	0,9757	0,97891	0,98493	0,97295	0,97704	0,97757	0,96075	0,98283	0,98333	47120,1

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = Tidak ada; Fungsi mutasi = *Uniform* dengan $P_m = 0,05$

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
1	Pop = 100	Pm = 0,05	Pc = 0,1	0,99677	0,99479	0,99618	0,99405	0,99724	0,99546	0,9933	0,99525	0,99548	0,99352	1,09524	0,99571	0,99772	0,99621	0,99713	48460,9
2			Pc = 0,2	0,99525	0,99285	0,99447	0,9966	0,99466	0,99488	0,99529	0,99466	0,99471	0,99644	0,99506	0,99587	0,99638	0,9958	0,99806	48117
3			Pc = 0,3	0,99626	0,99759	0,99444	0,99416	0,99449	0,99519	0,99591	0,99752	0,99438	0,99478	0,99548	0,99688	0,99635	0,99507	0,99615	48122,1
4			Pc = 0,4	0,99731	0,99677	0,9946	0,99611	0,99473	0,99439	0,99693	0,99595	0,99393	0,99531	0,99563	0,99452	0,99805	0,99637	0,99589	48126,9
5			Pc = 0,5	0,99564	0,9961	0,99458	0,99676	0,99544	0,99743	0,99541	0,9924	0,99528	0,99572	0,99402	0,9969	0,99623	0,99694	0,99482	48115,6
6			Pc = 0,6	0,99327	0,99755	0,9941	0,99459	0,99706	0,99514	0,99295	0,99627	0,99607	0,99288	0,99477	0,99725	0,9953	0,99475	0,99671	48107,1
7			Pc = 0,7	0,99634	0,99265	0,99396	0,99396	0,99353	0,99053	0,99363	0,99267	0,99578	0,99394	0,99362	0,9963	0,99655	0,98825	0,99666	48048,9
8			Pc = 0,8	0,9923	0,99097	0,99262	0,98613	0,99106	0,99267	0,99164	0,99045	0,99174	0,99254	0,99225	0,99344	0,99342	0,99226	0,99553	47962,4
9			Pc = 0,9	0,98492	0,99564	0,99217	0,97691	0,9811	0,98118	0,99379	0,99163	0,99076	0,98501	0,98426	0,98414	0,99261	0,98423	0,98862	47720,8

Paramater : $N_{pop} = 30$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba		x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
10		Pm = AF	Pc = 0,1	1	1	1	0.5911	1	1	1.5628	1	1	2	2	1	1	1	55178,6
11		Hybrid	Pc = 0,2	1	1	1	0.6226	1	1	1.5313	1	1	2	2	1	1	1	55178,6
12			Pc = 0,3	1	1	1	0.5518	1	1	1.6021	1	1	2	2	1	1	1	55178,6
13			Pc = 0,4	1	1	1	0.6046	1	1	1.5493	1	1	2	2	1	1	1	55178,6
14			Pc = 0,5	1	1	1	0.5789	1	1	1.575	1	1	2	2	1	1	1	55178,6
15			Pc = 0,6	1	1	1	0.6146	1	1	1.5393	1	1	2	2	1	1	1	55178,6
16			Pc = 0,7	1	1	1	0.5839	1	1	1.57	1	1	2	2	1	1	1	55178,6
17			Pc = 0,8	1	1	1	0.5955	1	1	1.5584	1	1	2	2	1	1	1	55178,6
18			Pc = 0,9	1	1	1	0.6497	1	1	1.5042	1	1	2	2	1	1	1	55178,6

Paramater : $N_{pop} = 50$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
10	Pop = 50	Pm = AF	Pc = 0,1	1	1	1	0,5637	1	1	1,5902	1	1	2	2	1	1	1	1	55178,6
11		Hybrid	Pc = 0,2	1	1	1	0,6132	1	1	1,5407	1	1	2	2	1	1	1	1	55178,6
12			Pc = 0,3	1	1	1	0,586	1	1	1,5679	1	1	2	2	1	1	1	1	55178,6
13			Pc = 0,4	1	1	1	0,579	1	1	1,5749	1	1	2	2	1	1	1	1	55178,6
14			Pc = 0,5	1	1	1	0,5409	1	1	1,613	1	1	2	2	1	1	1	1	55178,6
15			Pc = 0,6	1	1	1	0,5785	1	1	1,5754	1	1	2	2	1	1	1	1	55178,6
16			Pc = 0,7	1	1	1	0,6108	1	1	1,5431	1	1	2	2	1	1	1	1	55178,6
17			Pc = 0,8	1	1	1	0,5753	1	1	1,5786	1	1	2	2	1	1	1	1	55178,6
18			Pc = 0,9	1	1	1	0,5339	1	1	1,62	1	1	2	2	1	1	1	1	55178,6

Paramater : $N_{pop} = 100$; Fungsi *hybrid* = *fmincon*; Fungsi mutasi = *Adaptive Feasible*

No	Uji Coba			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	Total Profit
10	Pop = 100	Pm = AF	Pc = 0,1	1	1	1	0,5762	1	1	1,5777	1	1	2	2	1	1	1	1	55178,6
11		Hybrid	Pc = 0,2	1	1	1	0,5509	1	1	1,603	1	1	2	2	1	1	1	1	55178,6
12			Pc = 0,3	1	1	1	0,5585	1	1	1,5954	1	1	2	2	1	1	1	1	55178,6
13			Pc = 0,4	1	1	1	0,5699	1	1	1,584	1	1	2	2	1	1	1	1	55178,6
14			Pc = 0,5	1	1	1	0,5962	1	1	1,5577	1	1	2	2	1	1	1	1	55178,6
15			Pc = 0,6	1	1	1	0,5725	1	1	1,5814	1	1	2	2	1	1	1	1	55178,6
16			Pc = 0,7	1	1	1	0,5427	1	1	1,6112	1	1	2	2	1	1	1	1	55178,6
17			Pc = 0,8	1	1	1	0,547	1	1	1,6069	1	1	2	2	1	1	1	1	55178,6
18			Pc = 0,9	1	1	1	0,5965	1	1	1,5574	1	1	2	2	1	1	1	1	55178,6

LAMPIRAN C

KONDISI KEKINIAN MINIMARKET



Halaman ini sengaja dikosongkan

BIOGRAFI PENULIS



Penulis lahir di kota Jakarta Timur, DKI Jakarta pada 03 Mei 1994 ini merupakan anak kedua dari dua bersaudara. Penulis menuntaskan pendidikan dasar di SDN Dukuh Menanggal II Surabaya, SMP Negeri 22 Surabaya, SMA Negeri 15 Surabaya dan melanjutkan kuliah jurusan Sistem Informasi FTIf – Institut Teknologi Sepuluh Nopember (ITS) Surabaya dan terdaftar sebagai mahasiswa dengan NRP 5211100072.

Selama menjadi mahasiswa, penulis tergabung sebagai pengurus Himpunan Mahasiswa Sistem Informasi 2012/2013 sebagai staf Biro Kesekretariatan dan Badan Eksekutif Mahasiswa Fakultas Teknologi Informasi 2013/2014 sebagai Sekretaris Umum 1.

Penulis memiliki ketertarikan pada bidang optimasi sehingga membawanya untuk memilih bidang minat Sistem Pendukung Keputusan di Jurusan Sistem Informasi dan membuat tugas akhir yang berkaitan dnegan bidang minat ini.